■ PROJECTS ■ THEORY ■ APPLICATIONS ■ CIRCUITS ■ TECHNOLOGY

NUTS AND VOLTS

www.nutsvolts.com
November 2014

EVERYTHING FOR ELECTRONICS

Tools of the Trade

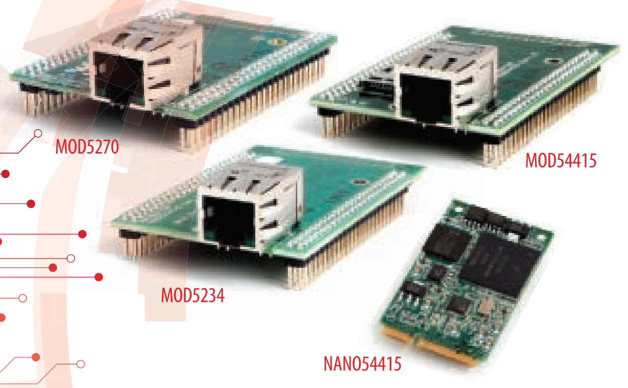
Having the right one for the job.

✦ PIC ✦ ARDUINO ✦ RASPBERRY PI

✦ Zener Diode Tester
✦ Accelerometer Data Logger
✦ Switch Your Switches To Pushbuttons

Ethernet Core Modules with

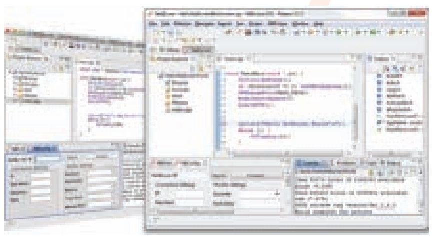
High-Performance Connectivity Options



- **MOD5270**
  147.5 MHz processor with 512KB Flash & 8MB RAM · 47 GPIO
  3 UARTs · I^{2}C · SPI
- **MOD5234**
  147.5 MHz processor with 2MB flash & 8MB RAM · 49 GPIO · 3 UARTs
  I^{2}C · SPI · CAN · eTPU (for I/O handling, serial communications, motor/timing/engine control applications)
- **MOD54415**
  250 MHz processor with 32MB flash & 64MB RAM · 42 GPIO · 8 UARTs
  5 I^{2}C · 3 SPI · 2 CAN · SSI · 8 ADC · 2 DAC · 8 PWM · 1-Wire® interface
- **NANO54415**
  250 MHz processor with 8MB flash & 64MB RAM · 30 GPIO · 8 UARTs
  4 I^{2}C · 3 SPI · 2 CAN · SSI · 6 ADC · 2 DAC · 8 PWM · 1-Wire® interface

Add Ethernet connectivity to an existing product, or use it as your product's core processor

The goal: Control, configure, or monitor a device using Ethernet



The method: Create and deploy applications from your Mac or Windows PC. Get hands-on familiarity with the NetBurner platform by studying, building, and modifying source code examples.

The result: Access device from the Internet or a local area network (LAN)

The NetBurner Ethernet Core Module is a device containing everything needed for design engineers to add network control and to monitor a company's communications assets. For a very low price point, this module solves the problem of network-enabling devices with 10/100 Ethernet, including those requiring digital, analog and serial control.

MOD5270-100IR........$69 (qty. 100)	NNDK-MOD5270LC-KIT..............$99
MOD5234-100IR........$99 (qty. 100)	NNDK-MOD5234LC-KIT............$249
MOD54415-100IR.....$89 (qty. 100)	NNDK-MOD54415LC-KIT..........$129
NANO54415-200IR...$69 (qty. 100)	NNDK-NANO54415-KIT...............$99

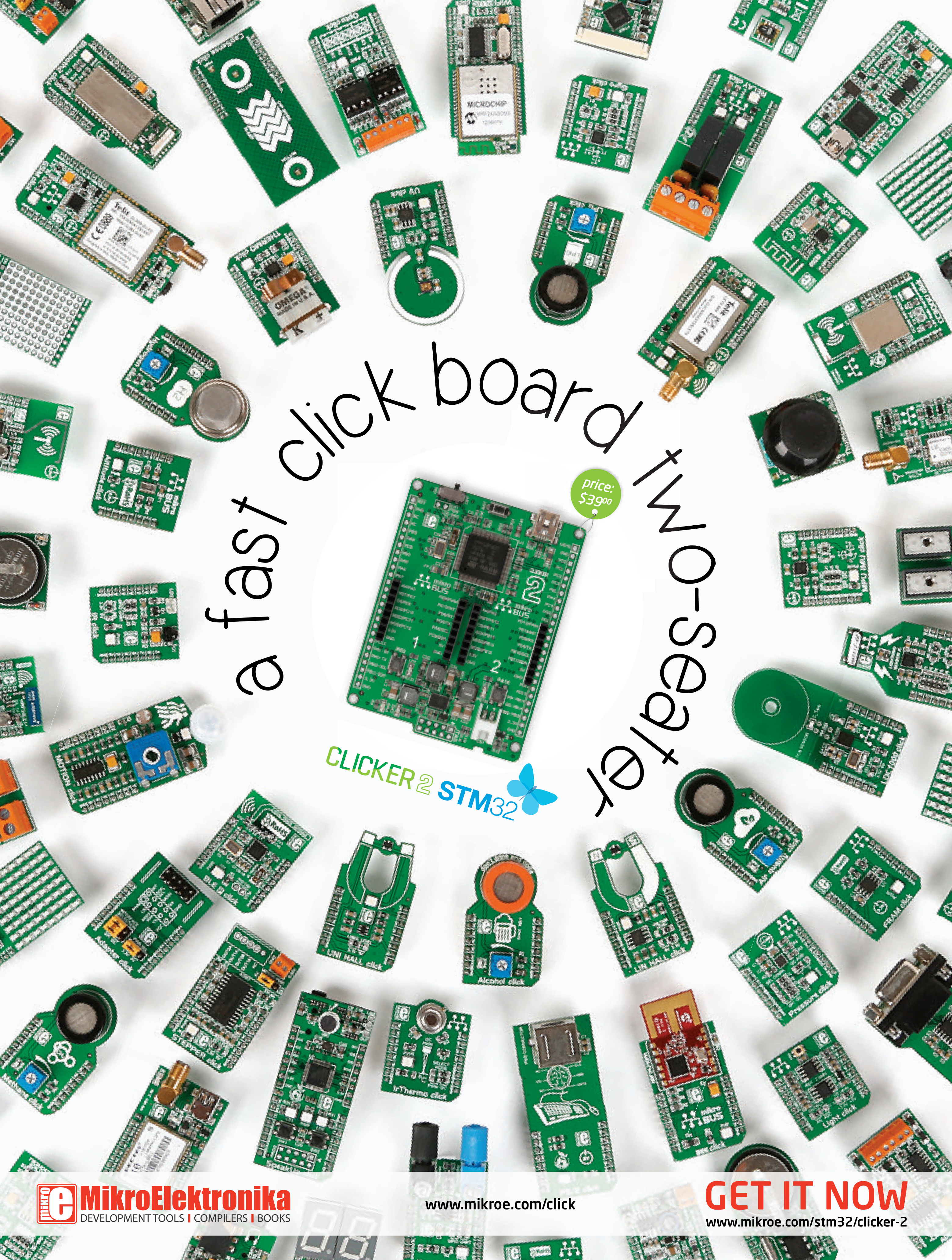
NetBurner Development Kits are available to customize any aspect of operation including web pages, data filtering, or custom network applications. The kits include all the hardware and software you need to build your embedded application.

For additional information please visit
http://www.netburner.com/kits

NetBurner
Networking in One Day!

Information and Sales | sales@netburner.com
Web | www.netburner.com
Telephone | 1-800-695-6828

freescale™
Alliance Member


a fast click board two-seater
price: $39^00
CLICKER 2 STM32
MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS
www.mikroe.com/click
GET IT NOW
www.mikroe.com/stm32/clicker-2

November 2014

Page 24

Columns

Departments

Nuts & Volts (ISSN 1528-9885/CDN Pub Agree #40702530) is published monthly for $26.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. PERIODICALS POSTAGE PAID AT CORONA, CA AND AT ADDITIONAL MAILING OFFICES. POSTMASTER: Send address changes to **Nuts & Volts, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@nutsvolts.com.

by Bryan Bergeron, Editor

DEVELOPING PERSPECTIVES

The Internet of Things

Since the birth of the Internet, there has been talk of total connectivity — between people, people and their possessions, and things to things. Up until recently, the reality has been that such ecosystems existed only in academic and corporate research centers. Today, the Internet of Things (IoT) is a practical reality in many settings.

Let's start off with consumer goods. For the well-heeled, there's the Philips hue connected bulb ($60). It's a wireless bulb that screws into a regular light bulb socket. The difference is that you can control color and intensity via your iPhone or iPad from across the room or across the globe. If light bulbs aren't your thing (they don't do much for me), then another example of a commercially available IoT device is the Nest Protect smoke and carbon monoxide detector. No more surprises when you return from vacation to find all of your belongings smoldering from a house fire.

Then, there's the Parrot flower pot moisture sensor ($60) so that you know exactly when to water your roses, even when you're away from home. I haven't yet made the move to keyless locks. The Kwikset Kevo wireless deadbolt ($220) — while expensive enough to suggest it isn't a toy — leaves me a bit uncomfortable, knowing that a hacker could unlock my front door from his desk in some other country.

I'm partial to the web-enabled wireless cameras that can be accessed from anywhere. Check who's at the front door, keep an eye on the sitter, make sure the kids are doing their homework — the list is almost endless. Sure, this sort of snooping has been available for years, but never so "plug and play" with a smartphone.

On the non-commercial end of things, I've had the pleasure of working with some expensive gear that has yet to trickle down to the consumer market. My favorite is the pallet tracker. It's an RF unit that not only broadcasts the GPS location of the tagged pallet, but informs the receiver about handling and environmental conditions.

For example, it can be set to sound an alarm if the internal accelerometer reads over 3 s, or, if the orientation of the tracker is shifted — as in flipped over. With such a device, it's possible to track a shipment across the globe, monitoring not only the location of the pallet but how it's been handled. (Be sure and check out the article this month by Ron Newton on a three-axis hockey puck accelerometer data logger that can be used for just such an application.)

I'm still waiting for a pair of running shoes that tell me not only how far and fast I've run — those have been available for years — but when it's time to buy a new pair of shoes based on impact sensors. It's the same with my toothbrush. I'm never certain exactly when to toss my brush for a new one. A connected brush that sends me an email after, say, 10,000 brushes would be helpful. I expect to see such a brush in my local CVS or Walgreens pharmacy within a year or two.

Best of all, I expect to see IoT kits that allow anyone to put anything on the Internet. Simply glue or nail it on whatever or whomever you want to track or control, and go about your business. One thing's for sure. With a mature IoT, I'll have to upgrade my data plan on my smartphone to avoid overuse charges. **NV**

NUTS AND VOLTS
EVERYTHING FOR ELECTRONICS

Published Monthly By
T & L Publications, Inc.
430 Princeland Ct.
Corona, CA 92879-1300
(951) 371-8497
FAX **(951) 371-3052**
Webstore orders only **1-800-783-4624**
www.nutsvolts.com

Subscription Orders
Toll Free **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277
North Hollywood, CA 91615

FOUNDER
Jack Lemieux

PUBLISHER
Larry Lemieux
publisher@nutsvolts.com

ASSOCIATE PUBLISHER
Robin Lemieux
robin@nutsvolts.com

EDITOR
Bryan Bergeron
techedit-nutsvolts@yahoo.com

VP OF OPERATIONS
Vern Graner
vern@nutsvolts.com

ADVERTISING SALES
Garry Moore
adsales@nutsvolts.com

CONTRIBUTING EDITORS
Joe Pardue, Fred Eady, Tim Brown, Chris Savage, Gordon Hoffman, Ron Newton, Jürgen Schmidt, Lou Frenzel, Ron Hackett

CIRCULATION DEPARTMENT
subscribe@nutsvolts.com

SHOW COORDINATOR
Audrey Lemieux

WEB CONTENT
Michael Kaudze
website@nutsvolts.com

WEBSTORE MARKETING
Brian Kirkpatrick
sales@nutsvolts.com

WEBSTORE MANAGER
Sean Lemieux

ADMINISTRATIVE STAFF
Debbie Stauffacher
Re Gandara

READER FEEDBACK

Programmed to Receive

When I got the August 2014 issue, I read Ron Hackett's work on programming the PICAXE. Thank you, Ron for taking the time to explain the ins and outs of the programming.

Lane

Project Strikes a Cord

Thomas Henry's October 2014 article on hot rodding your electric guitar was very interesting. I have done mods like this over the years myself and for the same reasons.

The biggest problem I have with this scenario is the idea of having batteries inside the guitar body. You will find that wood screws are not meant for iterative removals, and soon the rear cover on your Les Paul will not be retained.

I have tried two ways to get around this: (1) a separate battery box that is cord-connected to the instrument; and (2) also using the guitar cord as a power cord.

In the first scenario, I prefer to use a single battery supply and then split it into two polarities on the PCB (printed circuit board).

Henry's comment about not wanting to add parts to the PCB to have a split supply seems to be far less an evil to me than the overhead of having two batteries economically (either by part's cost or space usage), when you consider the maintenance issues of having two 9V cells in the guitar.

While you can use the classic

Continued on page 72

In this column, Tim answers questions about all aspects of electronics, including computer hardware, software, circuits, electronic theory, troubleshooting, and anything else of interest to the hobbyist. Feel free to participate with your questions, comments, or suggestions. **Send all questions and comments to: Q&A@nutsvolts.com**

- **Broken Laptop**
- **VCR Eats Tapes**
- **Sound System Hum**

Post comments on this article at **www.nutsvolts.com/index.php?/magazine/article/november2014_QA**.

Broken Laptop

Q I'm sad! My laptop went bad on me! I've never worked on one before, so my hope is that I can fix it with your help. It turns on and that is good. It comes on saying "Waiting for Windows," the mouse appears, then Waiting for Windows disappears. However, the mouse stays on the screen. What is that telling me? It's an old computer and I hear that laptops are not great to work on. If I had a clue what to look for, it would be interesting to try. Thanks.

— Frank G.

A All may not be lost. There are a couple of possibilities: a virus or the hard drive's "boot record" has been corrupted. The solution to both problems is similar. First, on the Start Menu, open the Run option and type in CMD to bring up the DOS Command Prompt (CP). [There is not enough room in Q&A for a full discussion of how to use the Command Prompt, so look on the Internet at websites such as **http://dosprompt.info** for more information.] In the CP box, you will see the Command Line C:\>; enter the commands here.

First, use the command CHKDSK/f/r to check the disk for bad sectors, recover readable files, and fix errors on the disk. If the laptop still will not boot, enter DIR in the CP and see if you can find any files that can be saved.

Keep any files you want by using the command COPY to save these files to another location (CD-ROM drive or external hard drive attached to the USB port or serial port). Be sure the files are saved on the new medium before proceeding.

The laptop either came with a startup disk (sometimes called a recovery disk), so insert the disk, boot the computer, and follow the instructions on the screen to reload the operating system onto the hard drive. If this doesn't work, the hard drive may need to be replaced which can be a hassle. You will need to get the manufacturer's instructions on how to open up the laptop and replace the drive (each laptop is different).

NOTE: If the laptop is really old and you have no files on it worth saving, you may want to buy a newer laptop (I am still using a 2003 PC and am on my third hard drive). I cannot overemphasize that you save your computer files (laptop or PC) to an external device so that you don't lose valuable data in the event of a hard drive crash. I learned years ago to save my files often in case of a power failure (I realize most software already does this, but there's no harm in being extra cautious).

VCR "Eats" Tapes

Q I have a VCR that has developed a problem of "eating" the tapes. I turn on the VCR, insert the tape, and press Play. After a few seconds, the VCR powers down, and when I turn the power back on the tape is stuck. Pulling on the tape to remove it results in a broken tape. I have a large collection of video cassettes and converting them to DVD format would be too expensive. What can I do?

— John D.

A A video cassette recorder (VCR) is a fairly complex electro-mechanical device consisting of a tape drive system, video drum with record/playback head, electronic controls, and sensors — all of which must operate synchronously to play the video tape housed in the cassette (early TV station video tapes were unwieldy reel-to-reel affairs which were NOT feasible for consumer use and were difficult enough for the station engineers).

We bought our first VCR in the mid '80s for approx $750 and it lasted about 15 years. We bought two VCRs in the early 2000s for about $350 (grandkids needed a separate TV), and they have lasted over 10 years (the last one finally gave up the ghost this year). Currently, VCRs can be bought for as low as $80.

As far as repairing any consumer electronic device, one rule I follow is if it is over five years old, you probably will not be able to find replacement components. Add to that the fact that most devices today have OEM chips — ICs that can ONLY be obtained from the manufacturer who makes more profit from selling new devices than repair parts — plus, the time and effort you expend on finding manuals (in the "old" days, they came with the device) for troubleshooting and repair.

Therefore, for the most part, it is more economical to

buy a new manufactured device than to try to repair an old one. Be sure to recycle the old device with your local community recycle system rather than sending it to the landfill via the garbage system. Die-hard electronics buffs can salvage parts from old VCRs (gears for robotics, capacitors, or even entire power supplies) which is the ultimate form of recycling: REUSE.

Sound System Hum

Q Our local theater group has me as its sound technician, even though I have no experience in this area. Our sound system consists of a main sound board, a monitor amplifier, house speakers, monitor speakers for the stage, and a series of wired and wireless microphones. The problem we are having is a 60 Hz hum on all of the monitor speakers.

We are using shielded cables from the wired mic to the main board; from the main board to the monitor board; and from the main board to the speakers. We have tried replacing the cables from the wired mics to their plugs and installing power line filters on the two sound boards (replacing the house cables will be fairly expensive and time-consuming), but this has not helped. Can you give me any ideas on how to solve this problem?

— Bill W.

A The 60 Hz signal originates from the power lines, but is not necessarily coming into your system directly from these. Since power filters on the incoming lines to the sound boards have not helped, this means the signal is not coming in directly which is a very rare occurrence. Unless you have shielded mic cables running close to a power line that is pulling heavy amperage, this hum is most likely not being introduced into the mic cables.

(Tip: Don't run low level signal cables near power lines, but if you must, run them perpendicular to the power lines instead of parallel to minimize picking up hum.) Since you have two line powered audio devices connected together (sound board and monitor amplifier), I suspect that you are experiencing a "ground loop."

A ground loop refers to a current in a conductor connecting two points in an electrical circuit that are supposed to be at the same voltage (ground, in this case) but, in reality, are at different voltages. Two signal ground points (main sound board amplifier output and monitor amplifier input) could have a small resistance between them so as to create a voltage drop due to magnetically coupled currents from the power lines in the signal ground conductor which causes the "ground current" to flow in the ground conductor which then causes a voltage variation in the input of a device such as the monitor amplifier.

In instrumentation signal lines, we were always told to disconnect the ground (shield) conductor at the load end (monitor amplifier), but audio equipment will give a terrible level of hum (if you have had a broken shield connection on a guitar cable, you know this is true), so that is not a good idea. Placing a small resistor (around 10 ohms) between the ground of the cable and monitor amp plug may reduce the hum to acceptable levels without inducing the "broken ground" problems.

My choice of solution is to insert an in-line isolation transformer into the audio line (in your case, at the monitor amp input). These devices fit right into the cable and amp input so there is no soldering or opening up the amp — just in case you are not an experienced electronics person (this is a not good time to start learning electronics).

These devices are $40 to $50, and you'll need to specify the type of connector input and output needed (phone, XLR, RCA, etc.). The sound systems I have worked with have a 1/4" stereo phone plug on the audio cable and a 1/4" jack on the input to the monitor amp. So, I would purchase a 1/4" stereo phone jack input on the transformer with a 1/4" plug on the output. There are many vendors in this magazine who can supply the isolation transformer you need, or check out a local electronics store. **NV**


EXPLAINING CES TO SOMEONE WHO HASN'T ATTENDED IS LIKE EXPLAINING PARTICLE PHYSICS TO A HOUSECAT.
In other words, you really can't. What you can do is make sure you don't miss what promises to be our most innovative and rewarding show ever. That and maybe take some pictures for the unfortunate cats left behind. Register at CESweb.org.
2015 International CES
THE GLOBAL STAGE FOR INNOVATION
PRODUCED BY CEA
JAN. 6-9, 2015
LAS VEGAS, NEVADA
CESWEB.ORG
#CES2015

BY RON HACKETT
www.jrhackett.net

Introducing the RazzPi PCBs

Before we dive into the content for this month's PICAXE Primer, there are a couple of things that I want to mention. First, in case you haven't heard about it yet, there's a new version of the Pi on the market: the Raspberry Pi Model B+. It's basically a re-designed Model B with several key improvements. I'll just list three of them to whet your appetite:

- **40 GPIO pins instead of 26**
- **Four USB ports instead of two**
- **Micro-SD card instead of a full size SD card**

If you're interested in learning more about the Model B+, there's a great Adafruit.com tutorial you may want to read. Just Google "Raspberry Pi Model B+ Tutorial" and you'll find it at or near the top of the list.

I also want to provide a brief update on my recent "reader feedback" request. At this point, I've heard from more than two dozen readers, and the results are still very similar to what I reported last time. Primer readers seem to be fairly evenly divided between those who would rather return to "pure" PICAXE projects, and those who want to continue our PICAXE-Pi explorations. So (as I mentioned last time) I'm going to continue writing about the topics that interest me — at least for the time being.

One reader (Dave Wreski) provided some important feedback on our UPS project (June 2014), which I want to mention here. He pointed out that I should have included a fuse for the battery pack because the D cells are capable of providing a significant amount of current. If the voltage regulator were to fail (and short out in the process), serious damage could occur. I think Dave's recommendation is a good one, and including an in-line fuse in the battery V+ line would be a simple matter.

Finally, a couple of readers have emailed to ask about the RazzPi printed circuit boards (PCBs) I mentioned back in the June column. (I had a problem with the first batch of boards that I received, so they weren't available on my site when I said they would be.) Fortunately, I recently received a second batch of the first three boards in the RazzPi series: RazzPi-LCD, RazzPi-20, and RazzPi-14.

The RazzPi-LCD is essentially a variation of the stripboard interface board that we have already used, with the added capability of attaching a standard HD44780 parallel LCD that's driven by six additional GPIO pins on the Pi. The RazzPi-20 and RazzPi-14 are complete PICAXE project boards that can be used with or without a Pi.

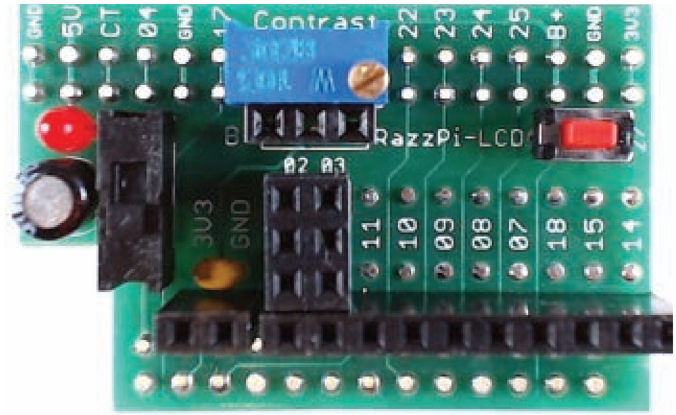
This month, we're going to take a look at two of the RazzPi boards. Even if you're not interested in PICAXE-Pi projects, you may still want to check out the RazzPi-20 for possible use in your pure PICAXE projects.

The RazzPi-LCD PCB

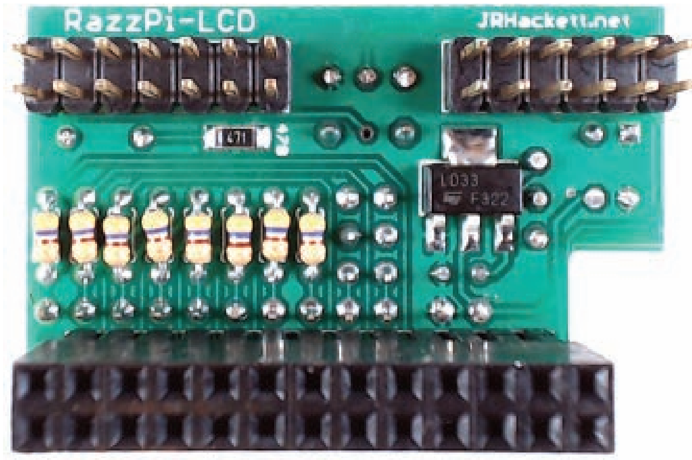
Figure 1 presents a top view of the final version of the RazzPiLCD PCB, which can be directly inserted into the Pi's 2x13 GPIO header. In **Figure 2** — which shows the bottom side of the RazzPi-LCD — you can see a 2x13 pin female header running along the lower edge of the bottom of the board. This header directly mates with the Pi's 2x13 pin male GPIO header, eliminating the need for a connecting cable. (As an aside, it should also be possible to use the RazzPi-LCD board with the new Pi Model B+ because the first 26 pins of the Model B+ 40-pin GPIO header are arranged identically to the older Model B GPIO header.)

In **Figure 2**, you can also see two separate headers at the top edge of the bottom of the RazzPi-LCD, where you would probably expect to see a single header for connecting to an external LCD. We're going to use a four-bit data connection for the external LCD and — as you may remember — that leaves four unused pins in the LCD's 16-pin header.

Rather than including the unused pins on the RazzPi-LCD, I chose to eliminate them to save some space on the board. (On the top of the

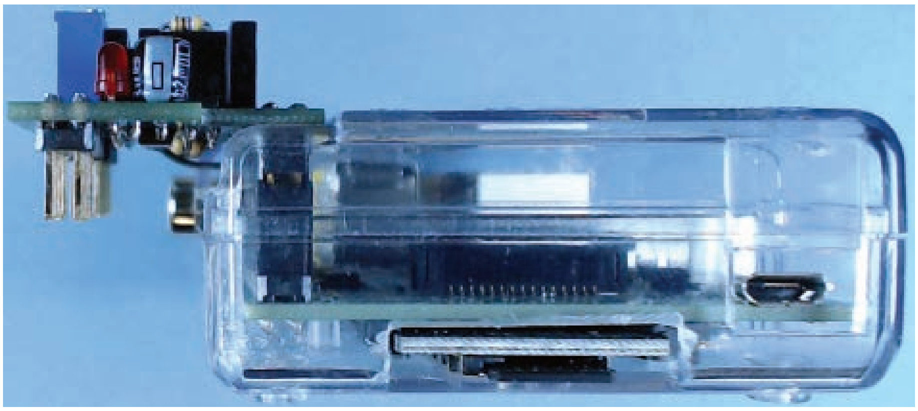


■ FIGURE 1. RazzPi-LCD, top view.



■ FIGURE 2. RazzPi-LCD, bottom view.


■ FIGURE 3. Adafruit "Pi Shell" case.

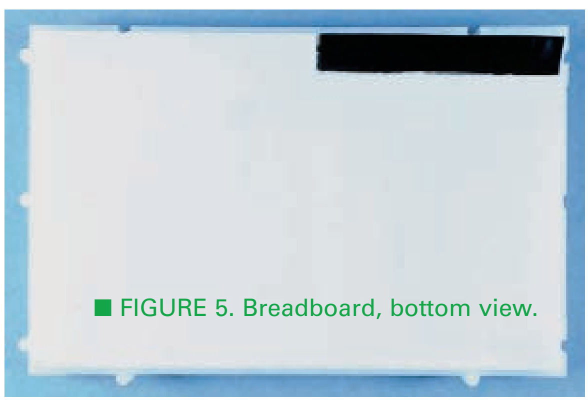

■ FIGURE 4. RazzPi-LCD and case, side view.

board, the variable resistor for the LCD contrast control occupies the space of the missing pins.)

Before we get into the details of the RazzPi-LCD circuit, let's take a quick look at how the board is connected to the Pi, to a breadboard, and to an LCD. To begin with — since the Razz-Pi LCD also interfaces with a breadboard — your Pi either has to be mounted in a case or you will need to add a plastic mounting plate to the top of the PI via its two mounting holes. (Obviously, using a case is the simpler approach.)

When I first designed the RazzPi-LCD, I decided to use the **Adafruit.com** "Pi Shell" case. This particular case works perfectly for the RazzPi-LCD because it includes a GPIO header opening that extends a little around the side of the case. In order to see what I mean, refer to **Figure 3** which shows how the GPIO opening wraps around on the Pi Shell case. **Figure 4** is a side view of my Pi installed in the Shell case, with a completed RazzPi-LCD inserted onto the Pi's GPIO header. As you can see, the top of the RazzPi-LCD is directly in line with the top of the case. This is necessary so that the breadboard can sit flat on top of the enclosure.

Figure 5 is a photo of the bottom of a breadboard that's been slightly modified to make sure that it does sit flat on the top of the case. As you can see, I removed a small section of the foam backing and covered the exposed power rails with electrical tape. When the breadboard is right side up (with the modification in its upper left corner), the modified area will sit directly on top of the short ends of the 2x13 pin


■ FIGURE 5. Breadboard, bottom view.

female header which is on the top of the RazzPi-LCD board. This allows enough room for the soldered pins, so that they don't push up on the breadboard.

I've been working on the RazzPi boards for a few months now. During this time, I purchased three Pi Shell cases for my three units. When I began writing this month's column, I went back to the **Adafruit.com** site to verify a couple of details about the Pi Shell case, only to find that it's been discontinued. Needless to say, I wasn't too happy about that! However, most plastic Pi cases are easy to modify so they can properly accommodate the RazzPi-LCD board. We don't have enough space to get into the details here, but I've included a thorough discussion of the necessary modifications in the RazzPi-LCD documentation that's available on my website (**www.JRHackett.net**).

Finally, if you refer back to the photo in **Figure 1**, you can see that the RazzPi-LCD provides access to all 17 GPIO pins on the Pi: 10 for the breadboard; six for a standard four-bit data connection to an HD44780-based LCD; and one for the onboard pushbutton switch.

Breadboard Connectors

The RazzPi-LCD simplifies the breadboard wiring because wire jumpers can easily be used to connect any of the 10 available GPIO pins directly to the breadboard circuit. I chose those specific 10 GPIO pins so that the Pi's three major communication protocols (I^2C, SPI, and serial I/O) are all available for connection to the breadboard circuit.

Similarly to the approach we used in our earlier stripboard interface board, the two I^2C lines (GPIO pins 2 and 3) include headers into which you can insert jumpers if you want to use the I^2C protocol, or two 470Ω current-limiting resistors if you prefer to use those two lines as general-purpose I/O pins.

The remaining eight GPIO lines all have 470Ω series resistors soldered in place on the bottom of the PCB. Finally, there is also a separate two-pin header for connecting +3.3V and ground to the breadboard's power rails.

External LCD Connector

As you probably know, the larger HD44780 LCDs include a standard 16-pin connector for backlit displays, or a 14-pin connector for non-backlit displays. We discussed the details of parallel LCD interfacing back in the April 2009 installment of the Primer, so I won't repeat them here. However, there are three points I do want to mention.

First, back in **Figure 1** you can see that each connection on the LCD header consists of a pair of pins rather than the single row header that's used on LCDs. I chose that arrangement because we're going to use a ribbon cable to connect to the LCD, and including both rows of pins (with each pair of pins electrically connected) ensures that either row of the IDC header on the other end of the ribbon cable can be connected to the LCD. Also, I needed the extra space for the contrast control resistor, anyway. As I mentioned earlier, there are no pins below the variable resistor, but the omitted pins are not needed in the four-bit data connection that we will use to drive an LCD.

Second, **Figure 6** presents the standard LCD 16-pin header connections, along with the corresponding LCD connections on the RazzPi-LCD PCB. The first thing you may notice is that **Figure 6** includes 17 (pairs of) pins rather than 16 (pairs of) pins; let me explain.

The first prototype of the RazzPi-LCD included a 2x16 pin male header for the LCD connection. However, when I tried to purchase 2x16 pin IDC connectors for the project, I couldn't find any! On the other hand, 2x17 pin IDC connectors are commonplace, so that's what I decided to use.

As it turns out, I'm glad I was forced into that choice because it provided the opportunity to include a 3.3V connection at pin 17. As a result, if a project doesn't require an LCD, the header on the RazzPi-LCD can be used to connect to an external breadboard, stripboard, or PCB circuit without requiring any additional power circuitry.

Of course, if you do connect the ribbon cable to an external LCD, you need to be extra careful to make sure you correctly line up pin 1 on the IDC connector with pin 1 on the external LCD. If you accidentally misalign the connection, you may damage the

RazzPi-LCD PCB Header	Pin	External LCD Header
GND	1	LCD Power Supply (Ground)
5V	2	LCD Power Supply (V+)
CT	3	Contrast Control (Analog)
GPIO 04	4	Register Select Input
GND	5	Read (Hi) / Write (Lo) Input
GPIO 17	6	Enable Input
* N.C.	7	** Data Bus I/O Bit 0
* N.C.	8	** Data Bus I/O Bit 1
* N.C.	9	** Data Bus I/O Bit 2
* N.C.	10	** Data Bus I/O Bit 3
GPIO 22	11	Data Bus I/O Bit 4
GPIO 23	12	Data Bus I/O Bit 5
GPIO 24	13	Data Bus I/O Bit 6
GPIO 25	14	Data Bus I/O Bit 7
B+	15	Backlight Anode (Optional)
GND	16	Backlight Cathode (Optional)
3V3	17	* N.C.
* No Connection		** Not used in four-bit mode

■ **FIGURE 6.** LCD interface pinout.

external LCD and/or the RazzPi-LCD board. One way to protect against that possibility would be to make an "LCD-only" cable by removing (or snipping) the two wires that would connect to pins 33 and 34 (which are both at the pin 17 position in **Figure 6**).

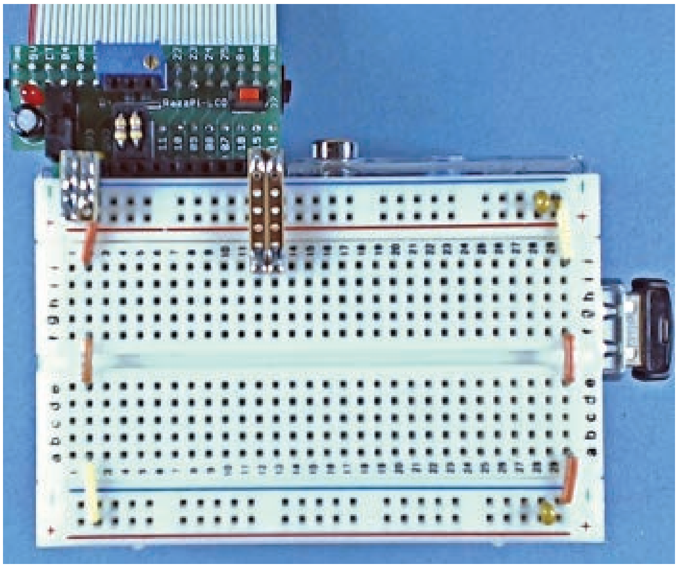
That way, the 3.3V line would not be connected to anything at the LCD end of the cable.

On the other hand, if you do connect the ribbon cable to an external circuit, you need to make a decision about whether you want to use the 5V or 3.3V power line, keeping in mind the fact that the Pi's I/O lines are 3.3V only. Also, don't forget that there are no current-limiting resistors on the available GPIO pins; they should be included in your external circuitry.

Additional Features

The RazzPi-LCD includes the same LCD contrast control and backlight circuitry that we used in our original parallel LCD project (April 2009). The 10K onboard potentiometer (which has different physical dimensions than the one we used earlier) can be adjusted for optimal contrast, and the three-pin female header just below it can be used to insert the appropriate current-limiting resistor for the LCD's backlight if it has one.

One final point about the LCD circuitry is worth mentioning. Most HD44780 LCDs are 5V devices, so it may seem strange that we're using the Pi's 3.3V level GPIO pins to directly drive the LCD. However, a 5V device always recognizes a 3.3V voltage level as a high level signal, so there's no problem with our direct connection. In addition, the fact that pin 5 in the LCD header — which is the LCD's read/write line (see **Figure 6**) — is permanently connected to ground means that we can only write data to the LCD; we can never read data, so there's no danger of the Pi receiving a 5V data signal from the LCD.



■ FIGURE 7. RazzPi-LCD breadboard setup.

Before moving on, I should briefly mention the RazzPi-LCD's momentary pushbutton switch that's connected to GPIO 27. In order to use the switch, your Python program needs to configure GPIO 27 as an input and enable its internal pull-down resistor. With that configuration, GPIO 27 will always be at ground whenever the normally open switch in not being pressed.

Whenever the switch is pressed, GPIO 27 is connected to the Pi's 3.3V power rail via the 470Ω SMD resistor on the bottom of the PCB, so the GPIO 27 input is pulled high. The onboard switch can be used for any purpose you want, but if you're running your Pi "headless" (i.e., no keyboard or monitor), the switch can provide an easy way to exit an infinite loop in a running program.

Now that we've covered the basics of the RazzPi-LCD board, the next logical step would be to discuss the process of assembling the board. However, I realize that many readers may not be interested in purchasing the RazzPi-LCD, so the detailed assembly instructions (along with a complete parts list and the RazzPi-LCD bare board) are available at my website.

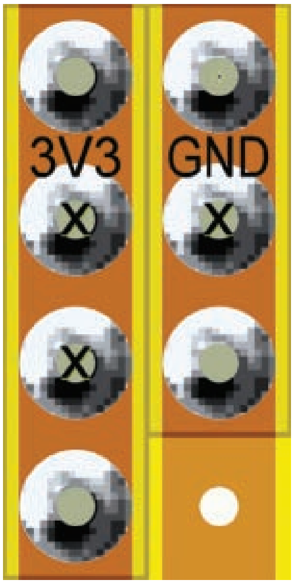
Interfacing an External LCD with the Raspberry Pi

The RazzPi-LCD greatly simplifies the task of connecting an external LCD to the Raspberry Pi. All you need are two 2x17 female IDC connectors and a piece of 34-wire ribbon cable. However, if you constructed the stripboard for our parallel LCD project in the April 2009 Primer, you could also use that board in conjunction with the PICAXE-Pi interface board we constructed in the August 2013 column. If you do that, don't forget that you will also need to supply +5V to the LCD.

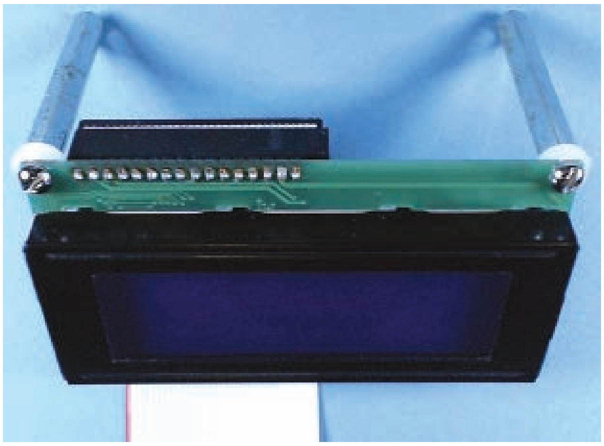
Figure 7 is a photo of a complete RazzPi-LCD breadboard setup. At this point, I should explain the two small stripboard circuits that you can see. On the first RazzPi-LCD system that I assembled, I removed the paper backing from a breadboard and affixed the breadboard in position on top of the Pi Shell case. However, I soon realized that I wanted the flexibility of moving a breadboard circuit from one Pi to another.

I tried just using wire jumpers to attach a breadboard to the RazzPi-LCD header, but the breadboard moved much too easily, sometimes bending the jumpers in the process. Also, the movement occasionally caused the power connection to be interrupted resulting in an accidental reset of the PICAXE circuit installed on the breadboard.

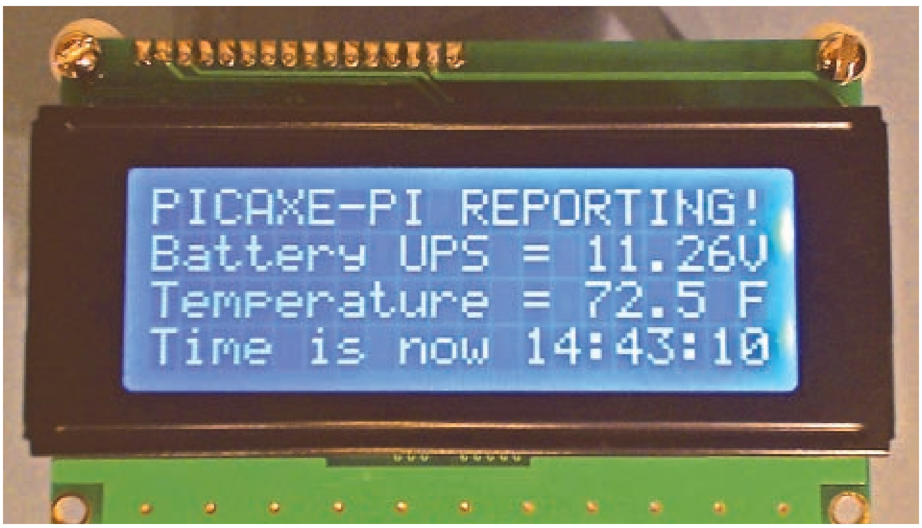
The obvious solution was to use the little stripboard "widgets" you can see in **Figure 7**. They make very solid and stable connections, and also hold the breadboard firmly in place. The short ends of the headers on both widgets are inserted from the top of the stripboard and soldered to the



■ FIGURE 8. RazzPi-LCD power widget.



■ FIGURE 9. LCD supports, top view.



■ FIGURE 10. *RazzPi-LCDtoLCD.py* display.

■ FIGURE 11. RazzPi-20, top view.

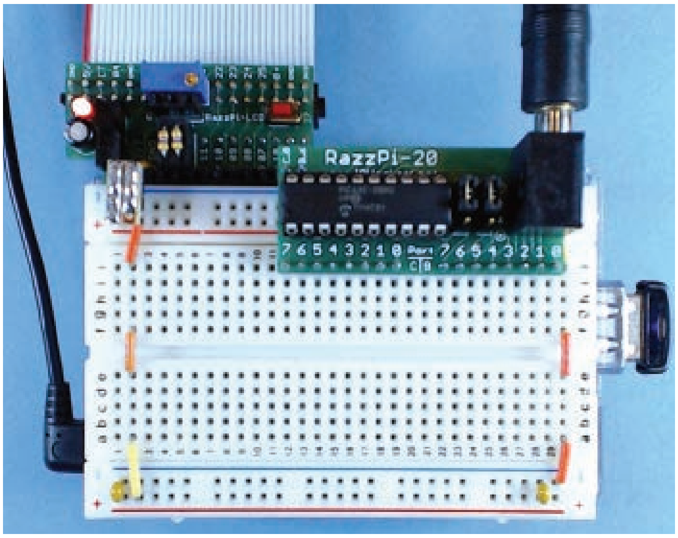
traces on the bottom; the widgets are used "upside-down" with their traces facing up.

The construction of the one that attaches GPIO pins 14 and 15 to the breadboard is obvious, but the power widget on the left does require a brief explanation. Since this is the only power connection to the breadboard, I decided to use multiple-pin headers so there are no open holes in the path of the power and ground signals. If you refer to the stripboard layout in **Figure 8**, you can see what I mean. The stripboard traces are facing up, and the headers are underneath the stripboard with all their pins soldered in place (four for the 3.3V connection, and three for the ground connection). Three pins in the photo have an "X" on them, which signifies that those three pins should be snipped off from the other side of the board.

When the widget is in use, the pins at A1 and B1 are inserted into the 3V3 and GND header on the RazzPi-LCD; the pin at A4 is in the breadboard's V+ rail (which is always below the ground rail in all our Primer projects); and the pin at B3 is in the breadboard's ground rail. As you can imagine, snipping the pin at A3 is critically important. If you forget to do so, you will have a direct short between the 3.3V and ground lines!

Finally, **Figure 9** is a top view that shows the two metal stand-offs that I'm using so that the LCD sits on my desktop at a comfortable viewing angle. (I happen to be using a 20x4 LCD, but you could do the same thing with a 16x2 LCD.) The IDC connector on one end of the 34-wire ribbon cable is inserted into the LCD header pins on the bottom of the RazzPi-LCD, and either row of the connector on the other end is inserted onto the LCD's 16- or 14-pin header. (Make sure that you correctly align pin 1 on the cable connector and the header pins.)

I used the *RazzPi-LCDtoLCD.py* program to produce the display shown in **Figure 10** (*RazzPi-*



■ FIGURE 12. Complete PICAXE-Pi setup.

Part	Label
Capacitor, .01 µF, 1206 SMD	.01 µF
Diode, 1N4148	4148
Header, male, two-pin (three pieces)	H1, H2, H3
Header, male, three-pin (two pieces)	C.0, B.6
Header, male, eight-pin (two pieces)	H4, H5
IC socket, 20-pin, machined	IC1
Resistor, 1K, 1206 SMD	1k
Resistor, 10K, 1206 SMD	10k
Resistor, 22K, 1206 SMD	22k
Stereo jack (tall)	Stereo jack

■ **FIGURE 13.** RazzPi-20 Parts List.

LCDtoLCD.py). It's available for downloading at the article link along with the second program (*Cylon15.bas*) that we'll be using this month. The *RazzPi-LCDtoLCD.py* program is fairly lengthy and may seem a little daunting at first, but the documentation that's available on my website includes a fairly thorough discussion of the most important points. By making minor changes in the variable and pin definitions, you can also use the *RazzPi-LCDtoLCD.py* program with the Pi stripboard interface that we developed earlier.

The RazzPi-20 PCB

The RazzPi-20 PCB is a complete PICAXE project board that accepts either a 20X2 or 20M2 processor (see **Figure 11**). Two of the processor's I/O pins (C.0 and B.6) are connected to the Pi's GPIO pins 15 and 14, respectively. As a result, the PICAXE processor can communicate serially with the Pi, without any additional wiring. I chose to connect those two PICAXE I/O pins so that a 20X2 or 20M2 processor will be able to use their *hserin* and *hserout* commands to talk to the Pi.

Figure 12 is a photo of a complete PICAXE-Pi breadboard setup, including the RazzPi-LCD and RazzPi-20 boards. In the figure, you can see how the processor's *hserout* pin (C.0) is connected to the Pi's *RxD* pin (GPIO 15), and its *hserin* pin (B.6) is connected to the Pi's TxD pin (GPIO 14).

All PICAXE M2-class processors also support the *hserout* and *hserin* commands, but the corresponding 20X2 commands include more powerful features. Also, eight of the Pi's GPIO pins remain available for breadboard circuits, and there's a good amount of room for installing peripheral components, as well.

Before moving on, there's one final point I want to mention about the *hserout* and *hserin* commands. You don't have to use them. If you feel more comfortable with the more traditional (and simpler) *serout* and *serin* commands, they also can be used with pins C.0 and B.6, respectively.

As I mentioned at the beginning of this article, if you're not interested in PICAXE-Pi projects, the RazzPi-20 (and RazzPi-14) boards can be used in pure PICAXE projects, as well. Refer back to the photo in **Figure 11** and note the two three-pin headers just to the left of the programming connector.

In the photo, each three-pin header has a two-pin jumper that's connecting the header's upper two pins. When the jumpers are in this position, pins C.0 and B.6 are connected to the two pins in the upper-left corner of the RazzPi-20. Since the two upper-left pins can be inserted into the GPIO 15 and 14 positions on the RazzPi-LCD header, this is the way to configure the Razzpi-20 when you are using it in conjunction with the RazzPi-LCD board and a Pi.

On the other hand, if you want to use the RazzPi-20 board in a pure PICAXE setup, all you need to do is move the two two-pin jumpers down so they connect the lower two pins on each of the three-pin headers.

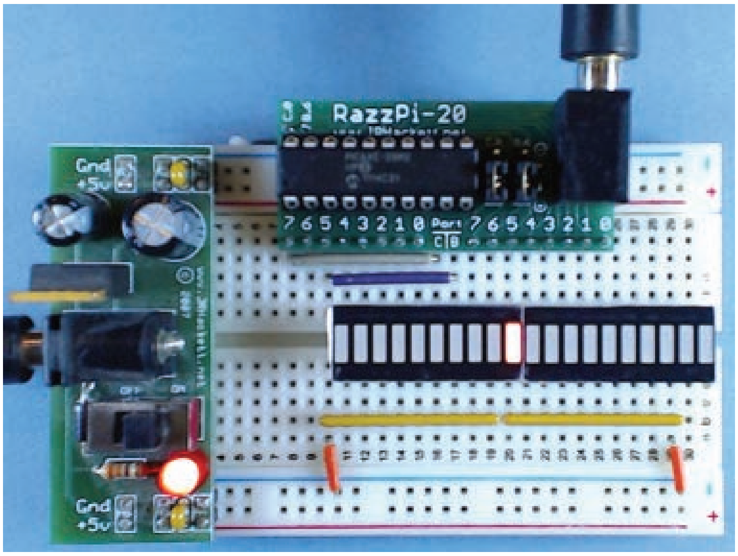
In that configuration, pins C.0 and B.6 will be connected to the appropriate positions on the port B and port C headers at the lower edge of the RazzPi-20 board.

Before moving on, a few additional points are worth mentioning about the PICAXE-Pi vs. pure PICAXE configurations of the RazzPi-20:

- If you're using the RazzPi-20 in a PICAXE-Pi project that involves serial communications, don't forget that pins C.0 and B.6 will **not** be available on the port B and port C headers.
- In PICAXE-Pi projects, you aren't limited to serial communications. By connecting the necessary RazzPi-20 pins with the appropriate pins on the RazzPi-LCD breadboard header, you can easily implement I^2C or SPI communications as well.

• If you use the same RazzPi-20 board in both PICAXE-Pi and pure PICAXE projects, make sure the jumpers on the two three-pin headers are in the correct position for your current project.

• If you never intend to use the RazzPi-20 in a PICAXE-Pi project, you could assemble the board without the two three-pin headers, and replace each one with a soldered-in wire jumper that connects the two lower positions for each three-pin header.



■ FIGURE 14. RazzPi-20 pure PICAXE setup.

RazzPi-20 Assembly Instructions

Since the RazzPi-20 board can be used with both PICAXE and PICAXE-Pi projects, let's take a look at the process of assembling the PCB. A complete parts list is shown in **Figure 13**, and all the parts (except the PICAXE processor) are available on my website. (PICAXE processors are available at **www.phanderson.com** and **www.sparkfun.com**.) As usual, read through the following list of assembly instructions to be sure you understand the entire procedure before beginning to put the board together.

1. On the top of the board, install the diode (observing polarity), the two three-pin headers, and the stereo jack. (**Do not** install the IC socket yet.)

2. Snip excess leads on the bottom of the PCB.

3. On the bottom of the board, install the SMD cap and the three SMD resistors.

4. Insert headers H1, H4, and H5 into the middle area of a breadboard so their positions match the spacing on the PCB layout.

5. Place the PCB on the three headers so that the short ends of all the pins are inserted into the corresponding holes in the PCB.

6. Solder headers H1, H4, and H5 in place.

7. Remove the PCB from the breadboard and insert headers H2 and H3 into the upper power rails of the breadboard so their positions match the spacing on the PCB layout.

8. Place the PCB on headers H2 and H3 so that the short ends of the header pins are inserted into the corresponding holes in the PCB.

9. Press down on the PCB so all the headers are fully inserted into the breadboard.

10. Solder headers H2 and H3 in place. (Header H2 will require extra care because the available space is very limited.)

11. Remove the PCB from the breadboard, insert the IC socket from the top of the PCB, and solder it in place on the bottom of the board.

12. Clean the top and bottom of the PCB.

Experiment 1: Implementing a Pure PICAXE Setup With the RazzPi-20

In this experiment, we're going to implement a 15 LED "Cylon Eye" project as a simple demonstration of how to use the RazzPi-20 board in a pure PICAXE project. **Figure 14** is a photo of my breadboard setup for the experiment. In addition to the power supply and the RazzPi-20, the only other components we need are two 10-bar LED displays, two 10-pin SIP resistors (nine 470Ω resistors each, with a common ground pin), and four jumper wires.

The setup is simple. The two LED bar displays are lined up so that the anode of the left-most LED is connected to pin C.4 on the RazzPi-20, and the two SIP resistors are installed so that their ground pins are at opposite ends and connected to ground with a short jumper. (The ground connection on the left end is one breadboard position to the left of the first LED.)

In **Figure 14**, you can also see how the two slightly longer jumpers connect pins C.7 and C.5 on the RazzPi-20 to LEDs #6 and #7 (counting from the left), which would otherwise not be connected to a 20M2 output pin. Of course, we can't use pin C.6 for this experiment because it's fixed as an input. As a result, we're using eight pins on port B, and seven pins on port C to implement our 15 LED Cylon Eye.

Finally, **Figure 14** demonstrates the amount of space the RazzPi-20 saves on a breadboard. Without it (or a similar PCB), there would be no way we could install a 20-pin processor and two 10-LED bars on the small breadboard we usually use in our projects.

When you've completed your breadboard setup, download the software for this experiment (*Cylon15.bas*) to your 20M2, and have fun watching the Cylon Eye scan back and forth!

When you tire of doing that, I want to clarify one point about the

Cylon15.bas program. If you read through the program listing, you'll see that I've used the PICAXE *lookup* command to implement our scanning. If we were just doing an eight-LED Cylon Eye on port B, there are other ways to accomplish the same thing. For example, consider the following code snippet:

```
for index = 1 to 7
  pause abit
  outpinsB = outpinsB * 2
next index
```

If we replaced the first *for/next* loop in *Cylon15.bas* with the above code, the program would be functionally identically. We could also use the same approach to implement the last *for/next* loop in *Cylon15.bas* as well, by dividing (rather than multiplying) *outpinsB* by two. However, it would be extremely difficult — if not impossible — to use the above approach to implement the scan on port C, because the port C pins are not in numerical order. Fortunately, the *lookup* command solves this problem. All we need to do is arrange the values in parentheses so that they match the physical order of the port C pins.

Implementing an RPi / RazzPi-LCD / RazzPi-20 System

We don't have enough space to include a complete RPi / RazzPi-LCD / RazzPi-20 experiment this month, and I'm sure some readers wouldn't be interested anyway. Also, we've already covered the necessary basics back in the April 2014 installment of the Primer. In that article, we used two 08M2 programs (*tempSerialToPi.bas* and *tempSerialFromAx.py*) to communicate with the Pi.

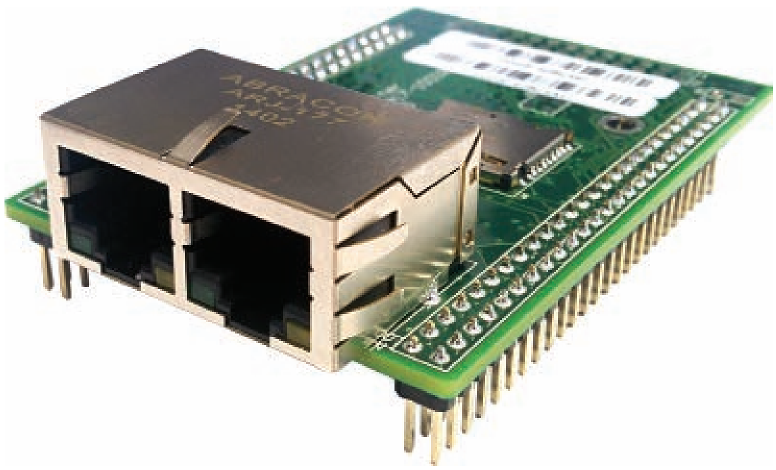
With minor changes in the pin and variable definitions, we could use those two programs as a starting point for any PICAXE-Pi project that uses the RazzPi-LCD and RazzPi-20 boards. In addition, we could also combine the functions of this month's *RazzPi-LCDtoLCD.py* program with the *tempSerialFromAx.py* program to enable the Pi to display the PICAXE data on the RazzPi-LCD's external LCD. Now that I think of it, that's a great programming challenge!

That's it for this month. If you're interested in the RazzPi boards, check out the additional information that's available on my website. In the meantime, keep those cards and letters coming, and have fun! **NV**

NEW PRODUCTS

- HARDWARE
- SOFTWARE
- GADGETS
- TOOLS

DUAL ETHERNET MODULE OPERATES AS INDEPENDENT PORTS OR SWITCH



The new NetBurner MOD54417 network core module provides 10/100 Ethernet connectivity with two Ethernet ports. The ports can operate independently — each with its own MAC address — or, as an Ethernet switch, simplifying network infrastructure (i.e., daisy chaining) by enabling Ethernet devices to connect through it.

The module is industrial temperature rated (-40°C to +85°C) and also provides eight UARTs, four I²C, two CAN, three SPI, 1-Wire®, a micro-SD Flash card socket, 42 digital I/O, eight 12-bit analog-to-digital inputs, two 12-bit digital-to-analog outputs, and five PWM outputs. Wireless 802.11 b/g/n communication is available with an optional Wi-Fi add-on.

The NetBurner Network Development Kit (NNDK) provides a complete software and tools package, including a real time operating system, full featured TCP/IP stack, Web server, DHCP server, Eclipse development environment, and C/C++ compiler and debugger.

The NNDK is focused on ease of use and enables users to have programs running within a few hours of receiving the kit. The price of the MOD54417 ranges from $94 to $129.

For more information, contact:
Netburner
Web: **www.netburner.com.**

TWO NEW TWEEZER METERS

Global Specialties has recently introduced two new tweezer meters: the LCR-58 tweezer LCR meter; and the PRO-57 tweezer digital multimeter. These meters are highly versatile/economically priced handheld meters designed to be small enough for one-hand operation on surface-mount devices (SMD). Perfect for use by engineers, technicians, electricians, and students, they have exceptional features and functions not offered by any similarly priced products. Much smaller and lighter than their comparable bench-type meters, these meters are high quality, precise, and full featured.

The LCR-58 LCR meter features include: basic accuracy of 1%-3%; resistance; capacitance; inductance; ESR; DCR; diode test; data hold; auto select mode; manually selectable frequency; and manual or auto range.

The PRO-57 digital multimeter features include: resistance; capacitance; continuity; diode test; data hold; auto select mode; and manually selectable range.

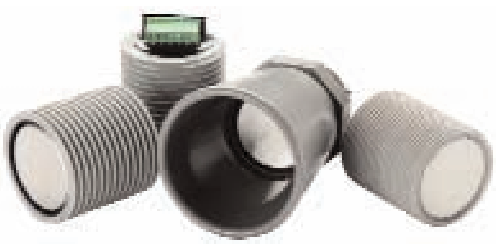
Available immediately, the LCR-58 has an MSRP of $99, while the PRO-57 lists for $39.

For more information, contact:
Global Specialties
Web: **www.globalspecialties.com**

If you have a new product that you would like us to run in our New Products section, please email a short description (300-500 words) and a photo of your product to:
newproducts@nutsvolts.com

HIGH PERFORMANCE 4-20 mA OUTPUT ULTRASONIC SENSOR

The new 4-20HR-MaxSonar-WR sensors from Maxbotix are high accuracy ultrasonic sensors featuring a 4-20 mA output. These sensors are a low cost IP67 rated drop-in replacement for use with existing PLC/process control systems. The sensors offer superior rejection of outside noise sources and feature speed of sound temperature compensation.



The 4-20HR-MaxSonar-WR sensors provide range information from 50 cm to 500 cm, and have a 1.6 mm resolution, an operational temperature range from -40°C to +65°C (-40°F to +149°F), real time automatic calibration, a 200,000+ hours MTBF, an operational voltage range from 12V to 32V, and a low 20-40 mA average current requirement. The sensors function well with multiple sensors in the same location, and they are RoHS and CE compliant.

A six-pin screw terminal header is included to simplify system connections for quick installation in applications such as: tank level measurement; tide/water level monitoring; solar/battery powered applications; industrial automation; and outdoor vehicle detection.

The 4-20HR-MaxSonar-WR sensors (and previous IP67 MaxBotix sensors) are manufactured in a variety of packages for easy mounting in existing fittings. The sensors are available in M30x1.5; 1″ BSPP; 1″ NPTS; and 3/4″ NPTS PVC pipe fittings. Pricing starts at $199.95 each and $134.37 in 100 unit quantities.

For more information, contact:
Maxbotix
Web: **www.maxbotix.com**

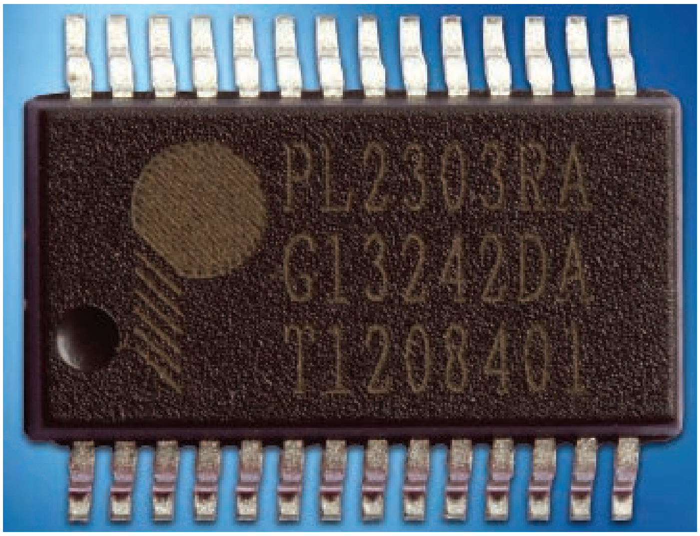
ALL-IN-ONE USB/SERIAL CONTROLLER

Saelig Company, Inc., now has available the Prolific PL2303RA — an all-in-one IC that provides a convenient low chip count solution for connecting an RS-232 full-duplex asynchronous serial device to any USB-capable host. This on-chip solution with its built-in charge-pump RS-232 transceiver and widely compatible virtual COM port drivers, and compact and low power serial devices can be made USB-ready easily.

The PL2303RA integrates a high baud rate RS-232 transceiver that meets the EIA/TIA-232F and CCITT V.28 communications interface specifications. The RS-232 transceiver is a three-transmitter/five-receiver design with a built-in high efficiency charge pump circuit. Adding external 1 µF capacitors provides the required bipolar output to the transmitters to deliver the correct RS-232 output voltage levels without requiring external active components. The PL2303RA operates at a very low power consumption with a guaranteed data rate up to 1,000 kbps, and also features an active-low shutdown indicator pin.

By taking advantage of USB bulk transfer mode, large data buffers, and automatic flow control, the PL2303RA is capable of achieving higher throughput than traditional UART (Universal Asynchronous Receiver Transmitter) ports. The flexible baud rate generator of the PL2303RA can be programmed to generate any data rate between 75 bps to 1 Mbps. It supports remote wake-up from any of its RS-232 input pin signals. Four general-purpose I/O pins are also available for use in product designs.

The PL2303RA has been designed with compact mobile and embedded solutions in mind; its small

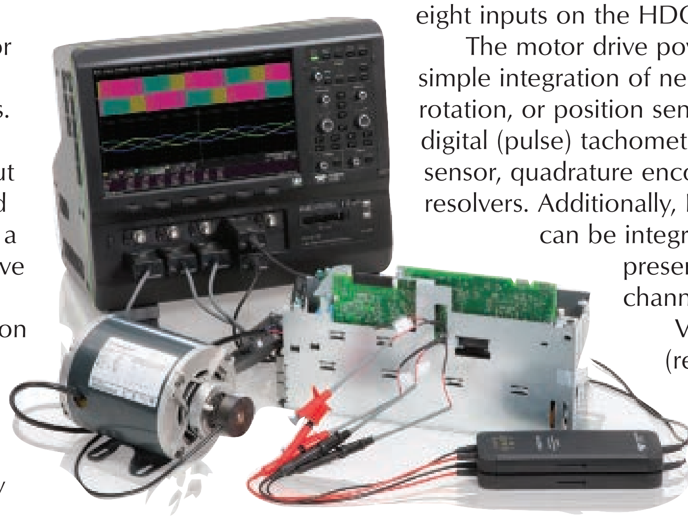


footprint fits easily into connectors and handheld devices. A complete USB-serial design can be accomplished without any external active components — just a few surface-mount resistors and capacitors. With its tiny power consumption in either operating or suspend mode, the PL2303RA is ideal for bus-powered operation, allowing plenty of spare power for any attached devices. It has been designed for -40°C to 85°C operation, and comes in a small footprint RoHS-compliant and Pb-free 28-pin SSOP IC package. The PL2303RA is manufactured by Prolific Technology, Inc.

For more information, contact:
Saelig Company, Inc.
Web: **www.saelig.com**

MOTOR DRIVE POWER ANALYZER SOFTWARE

Teledyne LeCroy now has available three-phase motor drive power analyzer software for its HDO8000 oscilloscopes. The HDO8000 mixed-signal oscilloscopes — with eight input channels, 12-bit resolution, and up to 1 GHz bandwidth — are a practical solution for motor drive embedded control and power section debug. With the addition of three-phase power analyzer capability and motor speed and torque integration, the complete drive system can be more quickly and easily validated and debugged, and extensive drive and motor power and efficiency measurements may be performed.



The HDO8000 and motor drive power analyzer software can be utilized by designers and manufacturers of motors, motor drives, variable frequency drives, variable speed drives, industrial automation, and motion control equipment. Design engineers that integrate motors into their designs (vehicles, power tools, appliances, elevators, fans, blowers, compressors, pumps, etc.) can utilize the product to validate proprietary controls and complete drive system designs.

A complete drive system is a complex mix of three-phase power electronics, motor/mechanical analog and digital sensors, and embedded controls, with a complex variety of analog, digital, serial data, and pulse-width modulated (PWM) signals. Oscilloscopes are able to capture higher speed embedded control system or power transistor activities, but until now have been unable to measure drive power and efficiencies.

Power analyzers are a single-use tool for measuring input/output "black-box" drive power and efficiencies, but provide limited or no waveform capture for embedded control or drive system debug.

The HDO8000 oscilloscope with motor drive power analyzer software permits waveform captures from the drive power section, individual power transistors, and embedded control system, and performs coincident three-phase power analysis of the power section waveforms in one high performance instrument, enabling debug and analysis of all aspects of the complete motor drive.

An intuitive graphical user interface provides setup/connection guidance. Setup capability is provided for any combination of single-phase or three-phase drive input/output, and supports both two- and three-wattmeter calculation methods and a line-line to line-neutral voltage conversion.

Using the two-wattmeter method, drive input/output and motor output efficiencies can be measured using the eight inputs on the HDO8000 oscilloscope.

The motor drive power analyzer software permits simple integration of nearly any type of speed, rotation, or position sensor, including analog and digital (pulse) tachometers, brushless DC (BLDC) Hall sensor, quadrature encoder interface (QEI), and resolvers. Additionally, Hall sensor and QEI signals can be integrated through digital inputs, preserving valuable analog input channels for other signals.

Various voltage, current, power (real, apparent, and reactive), phase angle/power factor, and efficiency parameters are calculated on acquired voltage and current waveforms and displayed in a table.

The table is user-configurable and is displayed along with the acquisition waveform, and corresponds to information normally provided by a power analyzer.

Additionally, a waveform showing any per-cycle measurement parameter variation can be displayed by simply selecting a table value. This waveform is time-correlated with other waveforms acquired by the HDO8000 oscilloscope and can be used to correlate complex drive behaviors to other control or power system waveforms, and to debug drive system problems. Statistical detail of the measurement set can also be displayed. This additional information goes well beyond what is provided by a power analyzer.

The motor drive power analyzer software retails for US $5,000. It will be available for all HDO8000 oscilloscope models (350 MHz, 500 MHz, and 1 GHz).

Beta versions of the new three-phase motor drive power analyzer software became available in October, with product shipments scheduled for January 2015. Any customer who has already purchased a Teledyne LeCroy HDO8000 oscilloscope may request the beta software.

For more information, contact:
Teledyne LeCroy
Web: **www.teledynelecroy.com**

Continued on page 71

ZENER DIODE TESTER

By Gordon Hoffman

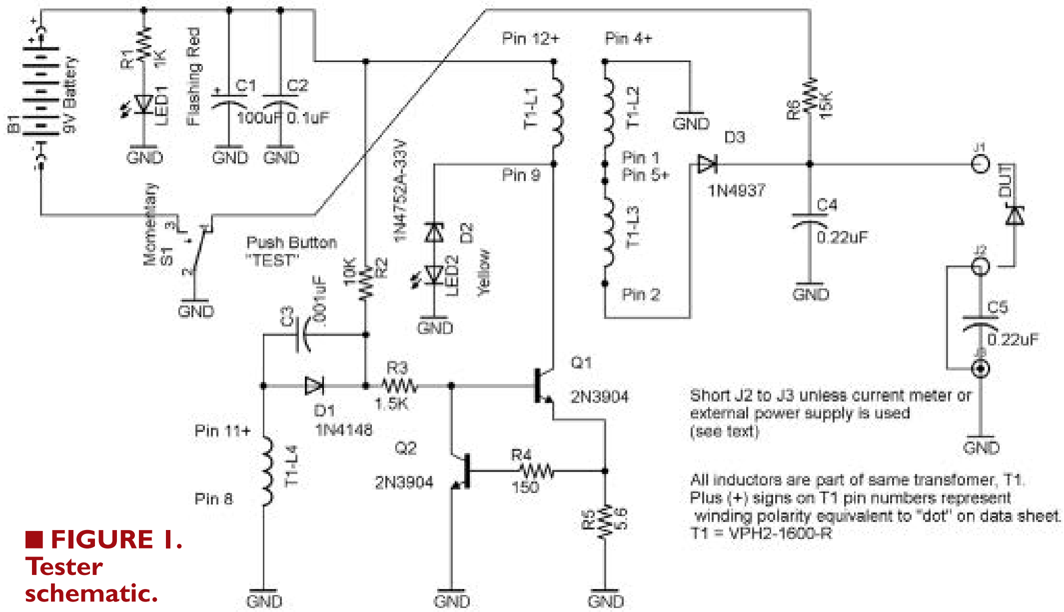
Post comments on this article and find any associated files and/or downloads at **www.nutsvolts.com/index.php?/magazine/article/november2014_Hoffman.**

When I pick a zener diode from my parts cabinet, I always want to check its breakdown voltage before I use it in a project — just to be sure there wasn't a mix-up in my cabinet drawer. This procedure needs to be repeated when putting unused diodes back into my cabinet, or when sorting among a bunch of diodes I picked up at a surplus outlet.

It's time consuming to set up a power supply and voltmeter, select a series resistor to limit current, and measure the voltage for each diode. Alternatives like reading the part number and looking it up, or setting up a curve tracer aren't any faster. Many errors, burned out diodes, and damaged projects have convinced me there must be a better way. While experimenting with self-oscillating switching power supplies for another application, it dawned on me that this technology was the answer to my problem.

The tester described in this article is a simple two-transistor circuit operating from a 9V battery which tests zener diodes with breakdown voltages up to 52 volts. Nothing more than a multimeter is necessary to build, test, and use the circuit. Its transformer-coupled design adjusts the output voltage automatically to the voltage of the zener diode under test, while adjusting diode current to preserve a relatively constant diode power throughout the measurement range.

There's no need to select a current-limiting resistor. Just connect a voltmeter across the diode and measure its breakdown voltage at a consistent and safe power level. The circuit will also safely test LEDs — including white LEDs that multimeters can't test and other low voltage diodes in their forward direction, as well as MOVs (metal-oxide varistors) and other protective devices with higher voltage breakdowns.



■ FIGURE 1. Tester schematic.

Connecting an external power supply to the tester allows devices that break down well above 50V to be tested easily, also. With all my projects and the numerous uncharacterized devices lying around my shop, this zener tester has proven to be the most regularly used equipment on my bench!

Zener Diode Background

If you're not familiar with zener diodes, then an introduction is appropriate. Zeners are so useful they should be found in every experimenter's component collection. The zener diode is a specialized silicon diode that looks like a normal silicon diode in the forward direction. However, in the reverse voltage direction, it exhibits a low leakage current like a normal diode until a voltage — the so-called "zener voltage" — is reached. At that point, it sharply exhibits a controlled constant breakdown voltage despite increasing current.

All diodes start to heavily conduct at some voltage in the reverse direction, but are operated well below this breakdown voltage to avoid damage. Zeners, on the other hand, are designed to be operated in their breakdown region, and that breakdown is carefully engineered for specific voltages. For example, the 1N4728 to 1N4764 family of zener diodes includes breakdown voltages from 3.3 volts to 100 volts in 37 steps — a much broader range than common fixed-voltage three-terminal IC regulators. These diodes can also be put in series to achieve almost any voltage desired, and different diode families come in various wattage ratings from 200 mW to over 10W.

Zeners are commonly used as voltage regulators, reference-voltage standards for op-amp designs, and as protection devices to safeguard components from over-voltage conditions. Applications like relay or solenoid drivers and switching power supplies — like the one described in this article — commonly use zeners for protection (like D2 in **Figure 1**).

They also find use (instead of a simple series resistor) in dropping one voltage down to a lower voltage, and in clipping AC voltage waveforms. They are even used as noise generators in RF impedance bridges. Best of all — particularly for experimenters on a budget — typical low power zener diodes cost just pennies each.

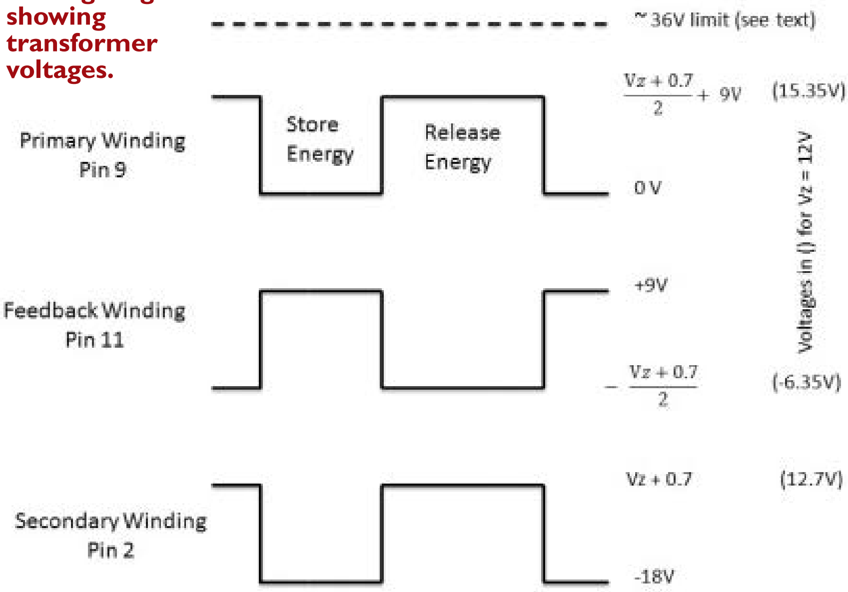
The Zener Diode Tester

The diode to be tested is connected between the red and yellow terminals, and the voltage is read across these terminals when the pushbutton is depressed to test. Connecting a current meter between the yellow and black terminals allows current measurement while testing if desired, or replacing the current meter with an external power supply allows extending the tester's range to well above 52 volts.

A flashing red LED indicates when voltage is present across the test terminals. A yellow LED indicates that the output is at the limit of 55V. When no diode is present or the zener diode has a breakdown voltage above 55V, this LED is illuminated. The LED also indicates correct functioning of the internal circuit for initial testing and serves as a safety reminder that a potentially dangerous voltage is present across the output terminals. When the test switch is released, the output is quickly discharged to zero volts for safety.

The internal circuitry of the tester is easily modified and simulated to change the voltage range or for use in other applications. These modifications and simulation

■ FIGURE 2. Timing diagram showing transformer voltages.



are discussed later in this article.

Circuit Description and Operation

The schematic of the tester is shown in **Figure 1**. The key to operation and building ease is the use of a transformer, T1, which is readily available from multiple distributors and designed for use in small switching power supplies. T1 has six independent and identical polarized windings of which four are used in the tester: T1-L1 is the "primary" driving the "feedback" winding, T1-L4, and storing energy in the transformer's magnetic field; then T1-L2 and T1-L3 are connected in series to form the "secondary," discharging the stored magnetic energy into the diode under test (DUT).

Some of you will recognize this basic circuit as a simple "blocking oscillator," widely used with vacuum tubes for radar applications in WWII, and later adopted as a transistor configuration for the earliest generation of solid-state switching power supplies. In the power supply realm, it's now termed a "flyback" circuit, harking back to CRT-based/TV horizontal-output circuits using this topology, or often referred to as a "boost" circuit. It's simple, works with many different transistors, and doesn't require specialized integrated circuits.

S1 is an SPDT momentary pushbutton switch. Until S1 is pushed, the 9V battery is disconnected from the circuit, and output filter capacitor, C4, is discharged through R8. Pushing S1 to the ON or TEST position removes R8 from the output, and applies 9V to the primary, T1-L1, and to the start-up resistor, R2. It also turns on LED1 — a flashing red LED that indicates to the user that the circuit is

PARTS LIST

NAME	DESCRIPTION	PART #	SOURCE
R1	1K		
R2	10K		
R3	1.5K		
R4	150		
R5	5.6		
R6	15K		
C1	100 µF/16V electrolytic		
C2	0.1 µF ceramic		
C3	.001 µF ceramic		
C4, C5	0.22 µF film	871-B32671P5224K000	Mouser
	EPCOS B32671P5224K000	(See construction text)	
D1	1N4148		
D2	1N4937 (fast recovery)	(See construction text)	
LED1	Flashing red	276-312	RadioShack
LED2	Yellow	276-350	RadioShack
Q1, Q2	2N3904		
T1	VPH2-1600-R	283-3671-ND	Digi-Key
	VERSA-PAC™ (Coiltronics)		
S1	SPDT momentary PB switch		
	Switchcraft 903X (pictured)	89K7683	Newark
	-or-	28063	Jameco
	-or-	275-1549	RadioShack
PCB		276-150	RadioShack
Enclosure	SERPAC 231 (Almond)	SR231A-ND	Digi-Key
		635-231A	Mouser

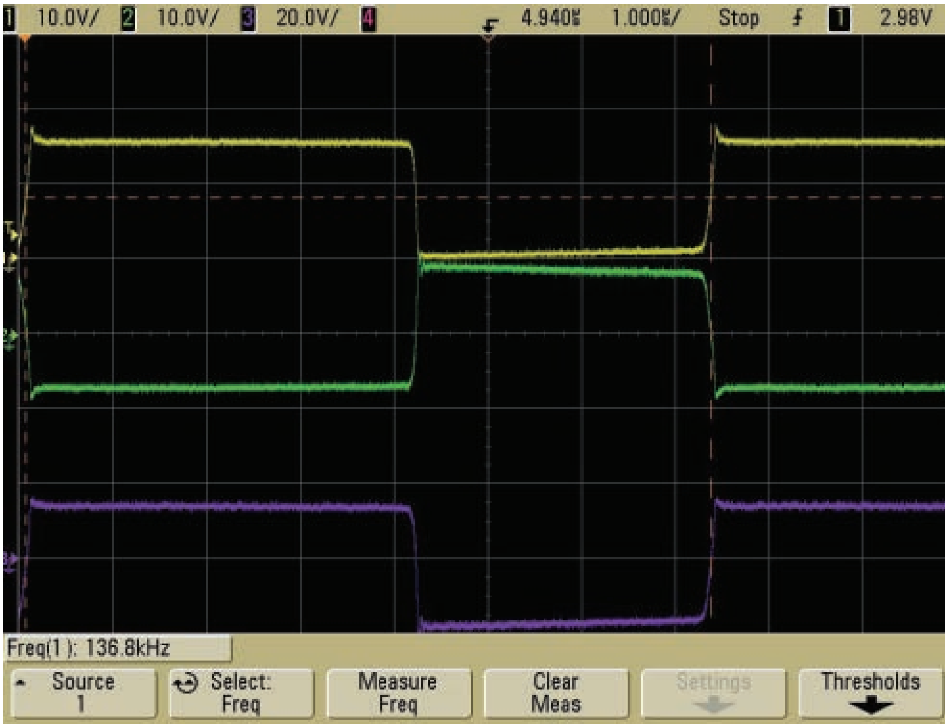
All resistors are 1/4W

energized and there are potentially dangerous voltages across the output terminals.

Start-up current flows through R2 (and R4), turning on drive transistor, Q1. As Q1 turns on, it pulls pin 1 of T1 toward ground which, in turn, causes the voltage across the feedback winding of T1/T1-L4 to rise from ground toward +9V, since the turns ratio of the two windings is 1:1. The rising voltage across T1-L4 is conveyed to the base of Q1 through C3, D1, and R3. This current adds to the current through R2, further turning on Q1 and rapidly driving it into saturation.

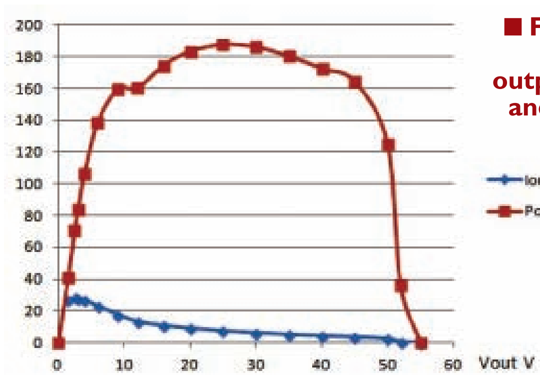
In the saturated state, the voltage across Q1 is a few tenths of a volt and almost the full 9V battery voltage is across T1-L1. Now, the current through T1-L1 and R6 begins to ramp up, storing magnetic energy in the core. No current flows through D3 since it's reverse-biased during this part of the oscillation cycle.

■ **FIGURE 3. Oscilloscope capture of actual circuit testing a 12V zener.**



When the voltage drop across R6 rises above 0.7V, the "choke" transistor, Q2, begins to turn on and shunt the base current of Q1 to ground, forcing Q1 to come out of saturation and the voltage at Q1's collector to rise. This action reduces the voltage across T1-L1, which correspondingly drops the voltage across the feedback winding, T1-L4, further reducing base drive to Q1 and shutting Q1 off rapidly through this regenerative action.

When Q1 comes out of saturation and begins to shut off, the voltage at its collector rises rapidly due to inductive action, and the voltage across T1-L1 reverses, driving Q1's collector voltage above 9V. At the same time, the secondary voltage reverses and D3 starts to conduct.



■ **FIGURE 4. Measured output power and current.**

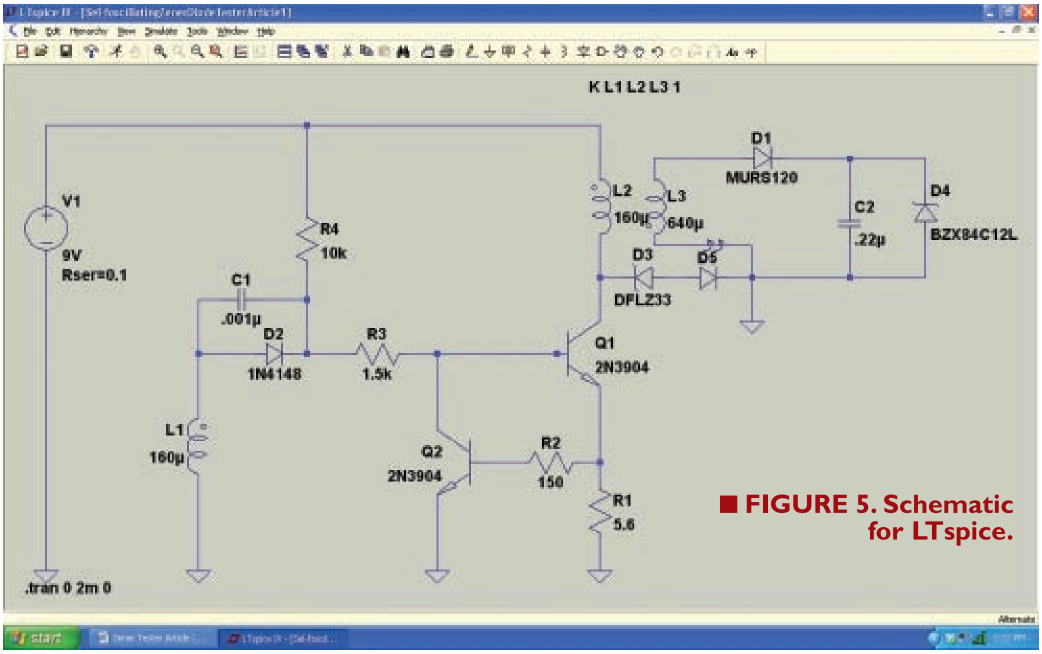
When the stored energy in the core is fully released through the secondary, the voltages on all windings collapse, turning on Q1 again via C1 (the voltage at pin 11 goes from a negative voltage toward ground). Then the cycle repeats until C2 is charged to a voltage level where the DUT begins to conduct, at which point the oscillation stabilizes and continues to feed power into the DUT.

Steady-state oscillation waveforms are diagrammed in **Figure 2**. Voltage levels (referenced to ground) are shown for a generalized zener diode voltage, Vz, under test. The voltages shown in parentheses are for a 12V zener diode as the DUT, and corresponding actual circuit waveforms are shown in **Figure 3**.

If no DUT is present when the circuit is operating, then the voltage across C4 will continue to rise, as will the peak voltage at the collector of Q1. The voltage at pin 2 of T1 and the output voltage would keep rising with each cycle, as would the peak voltage (half the output voltage plus 9V) at the collector of Q1. This is a feature of a flyback circuit configuration, which allows zener diodes to be tested well above the battery voltage of 9V.

However, some protection is necessary to keep the peak voltage at Q1's collector from exceeding its maximum collector voltage rating of 40V. The series combination of zener diode D2 and yellow LED2 provide this protection by limiting the peak voltage and absorbing the energy of T1's magnetic field if no DUT is present, or if the DUT breakdown voltage is greater than the tester's maximum output voltage. LED2 lights when there's current through D2 in this condition.

Figure 4 shows actual measurements of current and power for various DUTs on the tester as constructed.



■ **FIGURE 5. Schematic for LTspice.**

Measurement of these diodes with a direct current supply and the same currents produced identical results, so measurement accuracy is excellent. It should be noted that the transformer inductance tolerance is ±30%, so your results may differ.

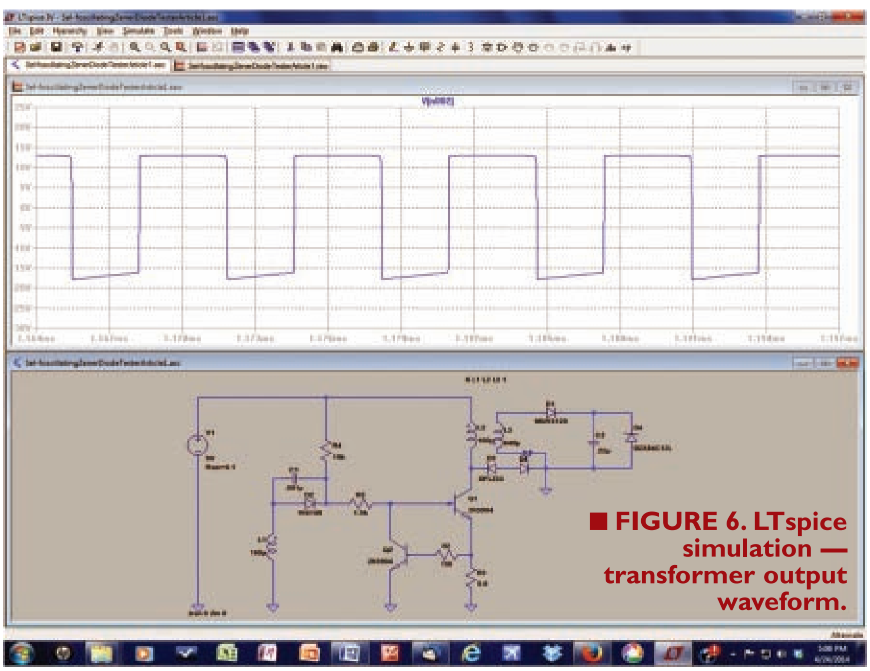
Simulation

Rather than attempt to explain the circuit's operation mathematically, it's easier to use simulation.

The free analog simulator from Linear Technologies — LTspice® (**www.linear.com/designtools/software/**) — is ideal for simulating this circuit and examining its detailed operation under varying component values and conditions. The simulator has virtual instrumentation that enables measurement of voltage, current, and power in every wire and component as a function of time.

Only components that strongly affect circuit behavior need to be modeled. The model circuit is shown in **Figure 5** with a 12V zener diode as the DUT. This file is available at the article link. A simulation screenshot showing output voltage waveforms from the output of the secondary winding (at connection to D1) is shown in **Figure 6**.

Components in the LTspice library were used, which differed in some cases from the actual components on the **schematic**. The transformer is modeled as a set of coupled windings with 100% coupling (K=1 in the Spice Directive for the transformer), and all inductances are assumed to linear without any current dependency. The actual transformer used specifies a 30% inductance decline at a current of 420 mA through a single winding — well in excess of the peak current in this design — so the assumption of linearity is reasonable. Simulated circuit behavior was very close to real circuit results and was particularly useful to optimize component values.

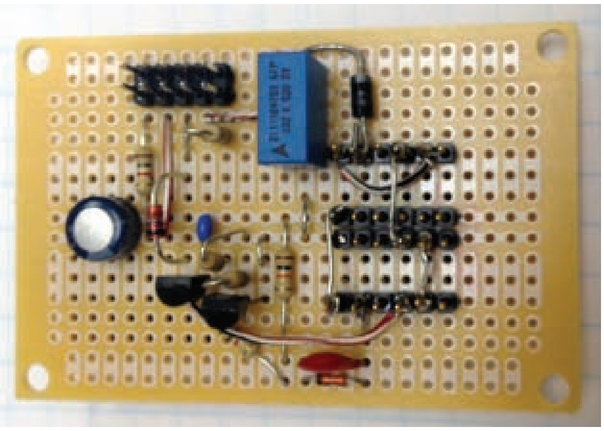


■ **FIGURE 6. LTspice simulation — transformer output waveform.**

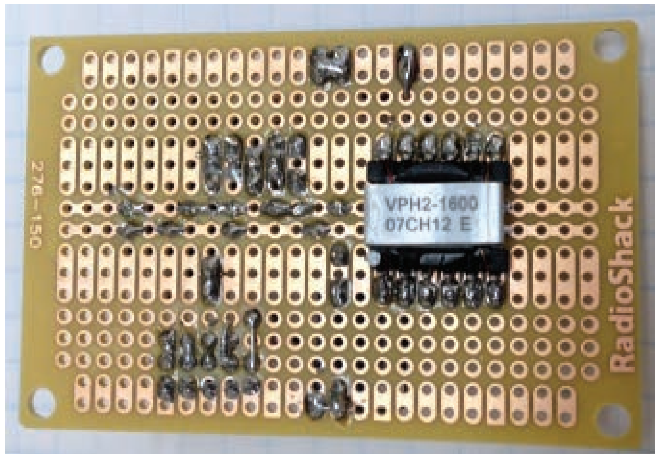
Construction and Testing

The circuitry is constructed on a prototype printed circuit board (PCB) from RadioShack, which also is a convenient fit into a standard plastic enclosure from SeraPac with a 9V battery compartment (see **Parts List**). The top of the board (**Figure 7**) contains all the components, except the transformer, T1, which is mounted on the underside (**Figure 8**). T1 is configured for surface-mounting, which fits nicely on the 100 mil centers of the PCB.

I used pin headers for the contacts to T1 and to connect to the front panel via a flat cable (10-conductor) with a connector



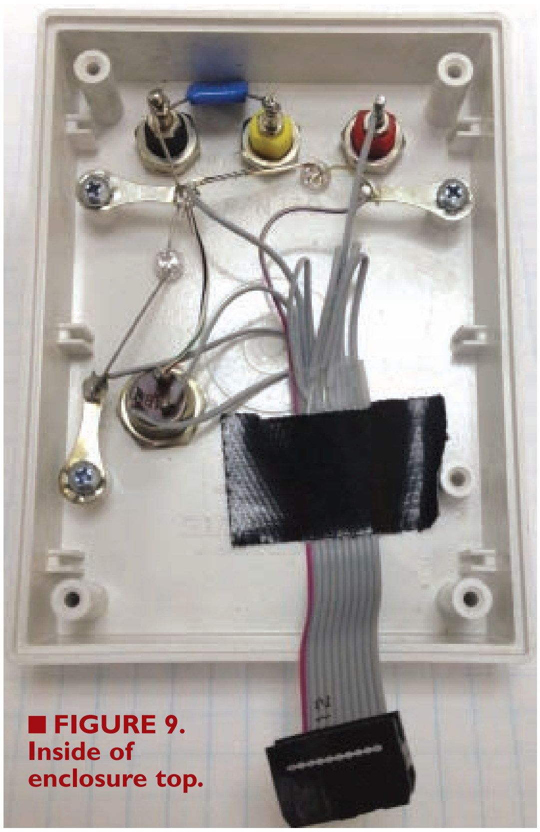
■ **FIGURE 7. Top of assembled PCB.**



■ **FIGURE 8. Bottom of assembled PCB.**

to the PCB (**Figure 9**). Neither is necessary, although I find pin headers convenient for securing oscilloscope probes when evaluating the circuit.

Having the front panel separate and easily disconnected also facilitates board construction and modification. The only caveat in using pin headers is to make sure the switch terminals of S1 don't contact the pin headers when the case top and bottom are put together.



■ **FIGURE 9. Inside of enclosure top.**

All the components on the board's topside should be mounted first and checked for continuity, and T1 soldered last on the underside. A final continuity check should be performed with the front panel connected.

Of particular importance is having D2 connected through LED2 to ground. If this connection is open, the voltage on the collector of Q1 can quickly rise to levels that will destroy the transistor.

When the continuity of the wiring checks out, connect a battery, leave the output terminals open (no DUT), and press S1. The yellow LED, LED2, should light, along with the flashing LED1. This is all you need to do to assure the circuit is operating. If the yellow LED doesn't light, recheck your wiring.

Aside from the transformer, most of the circuit components aren't critical but D3 must be a fast-recovery type rectifier — although any fast recovery diode with a breakdown voltage above 100 volts is acceptable. C4 and C5 should have a low equivalent series resistance (ESR) to avoid excessive ripple in the DUT. Film types are adequate in this regard, and the capacitors selected have an ESR under 0.1 ohms. If you're not sure of the ESR of the capacitors you have, parallel several of lower value (e.g., two 0.1 µF) and make sure they have an adequate voltage rating. The low value (0.22 µF) of these capacitors is adequate for measurement accuracy, while limiting the stored energy for safety reasons.

Operating the Tester

Short the yellow to the black terminal with a jumper, connect a zener diode across the red and yellow terminals (banded side to the red terminal) along with a voltmeter, and read the voltage across the diode after pressing S1. You'll note that as you hold S1 down, the zener voltage will drift up as the diode heats, so make a quick measurement.

Testing of LEDs and other low voltage diodes is also possible; just make sure the positive end of the LED or diode is connected to the red terminal to measure the forward voltage drop. Otherwise, the high voltage potential of the tester could destroy the LED or diode by exceeding its maximum reverse breakdown specification.

If you want to measure the current through a zener diode, remove the jumper between the yellow and black terminals and insert a millimeter. Capacitor C5 across these terminals provides a low impedance path to the pulsing current through the DUT so that the lead

inductance of the multimeter doesn't affect the readout accuracy.

When measuring an unknown zener diode and the yellow LED lights up, check to see if the diode is open by checking its forward voltage drop with a multimeter, or just reverse it in the tester. If the yellow LED goes out with the zener in the forward direction, then the diode is likely good but has a breakdown voltage above 55V. If you wish to measure the breakdown voltage in that case, insert an external variable power supply across the yellow and black terminals, with the minus terminal of the power supply connected to the yellow terminal. Increase the value of the supply slowly until the yellow LED is just extinguished, then measure the voltage across the diode.

I've measured zener diodes with breakdown voltages near 200V in this manner, as well as MOVs and other high voltage protective devices without concern for excessive power dissipation since the diode current when the yellow LED turns off is quite low.

Circuit Modifications and Improvements

The circuit configuration back in **Figure 1** is robust and will operate with a variety of modifications. The three variable elements you can experiment with are: the secondary winding of T1; the resistor R6, which determines peak current in Q1; and the breakdown voltage of D2.

If you want the output voltage to have lower maximum voltage, you can eliminate one winding in the secondary or reduce the breakdown voltage of D2. If you want a higher maximum voltage at the output, you can put a third winding (there are two unused windings on T1) in series with the two shown, or simply replace D2 with a higher voltage zener diode. If you take this latter route, then you'll need to pick a higher breakdown voltage transistor, like the MPSA06 (VCEO = 80V vs. 40V for the 2N3904).

Capacitors C4 and C5 are rated at 520V, and D3 has a reverse breakdown voltage of 600 volts, so there's room to play ... but be careful if you go to higher voltages. Although C4 is a small value (0.22 µF), energy storage goes up as the square of voltage, so higher voltages can deliver a very dangerous and potentially fatal jolt! Be careful!!

If you want to increase or decrease the power delivered to the DUT, then decrease or increase the value

of R6, respectively. Both 2N4401 and MPSA06 transistors can support peak currents up to 500 mA and are usable in this circuit.

It would also be easy to use a larger enclosure for the tester and include a digital panel meter that would read out the zener diode voltage without requiring a separate meter, or two to read voltage and current simultaneously.

There you have it! I hope this circuit measures up for you as much as it has for me! **NV**

References

Switchmode Power Supply Handbook, Keith Billings, McGraw-Hill, 1989, pp. 2.49 to 2.62.

VERSA-PAC Transformer Information: **www.digikey.com/product-search/en?mpart=VPH2-1600-R&vendor=283**

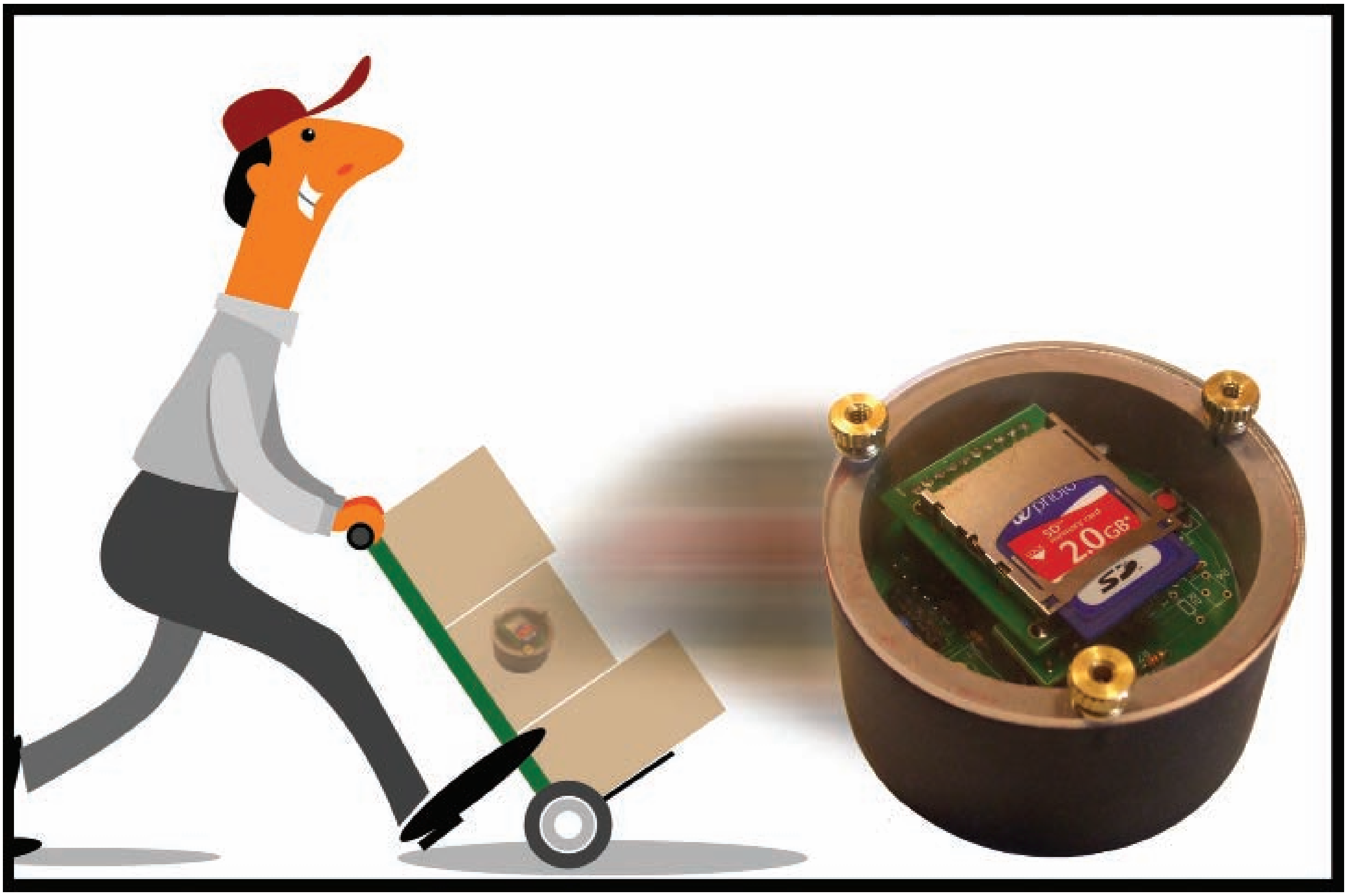
EDN Magazine, June 10, 2010, Design Ideas, pp. 51-52, "Circuit lets you measure zener voltages and test LEDs."

EDN Magazine, November 25, 2004, pp. 104-106, "Zener test circuit serves as a DC source."

THREE-AXIS HOCKEY PUCK ACCELEROMETER DATA LOGGER

By Ron Newton

Post comments on this article and find any associated files and/or downloads at **www.nutsvolts.com/index.php?/magazine/article/november2014_Newton.**



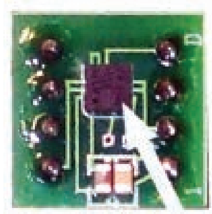
Recently, I've had a lot of email activity on my "Poor Man's Seismometer," published in the *Nuts & Volts* May 2012 edition — especially since geophones became available for $15 on eBay. Geophones increase the sensitivity and make for a small unit. So, I decided to follow up with a three-axis accelerometer, which is an adjunct for seismology.

Accelerometers can be used in many applications. For those who are studying vulcanology, accelerometers can be used for determining which direction the magma is moving using the Z axis. The sensor being utilized measures in three dimensions, using an ST Microelectronics 2-16 g LIS3DSH accelerometer.

In my first design, I used a Freescale Semiconductor 2-8 g three-axis accelerometer eight-bit A/D. However, the ST model (which is 16 bits) is 256 times more sensitive, and has a FIFO register which allows you to see activity just prior to an earthquake. ST also makes a 100 g to 400 g sensor which plugs in, replacing the lower g unit if you are interested in impact gs or a shipping monitor.

■ **FIGURE 1.**

■ **FIGURE 2.**



■ **FIGURE 3.**

I designed this project so that it can be programmed utilizing the same SD card as the previous seismograph article used. **Figure 1** shows the low g accelerometer which can be used as a seismograph, continuous monitor, or a tripping monitor. The data collected on the SD card can be downloaded into a computer and displayed using either Excel or the Dataq web browser (which is free software from Dataq). All three dimensions are displayed at the same time, giving the date, time, and parameters.

There are two main types of scales for measuring earthquakes: the Richter and the Mercalli. The Richter scale is logarithmic and was developed in 1935. Each number represents 10 to that power. An earthquake of three is about what most people can feel: 10^3 = 1000. The San Francisco, CA quake in 1906 was estimated to be an eight: 10^8 = 100,000,000. This is 100,000 times more powerful than a three.

The Mercalli scale is an arbitrary measurement based on what people feel. It was revised in 1902, and is a 12-point scale. The San Francisco quake equates to an X or XI on the Mercalli.

Gravity is the force pulling things down, and accelerates objects 9.8m/s^2. For something to leap in the air during an earthquake requires a force wave traveling above 9.8m/s^2 (1 g) which is equivalent to an 8.1 on the Richter, or an 11 on the Mercalli scale.

The sensor used in this article can measure down to .00006 per digit, or 0.00006m/s^2 (**Figure 2**). This would be a 2-3 on the Mercalli scale or about a 2 on the Richter scale. The accelerometer is more akin to the Mercalli scale than the Richter scale. What the accelerometer does over the seismograph is to give you a three-dimensional view of the g forces involved. This is an engineer's dream tool for studying structural engineering.

In the seismograph mode, once an earthquake above .001 g is detected on the X or Y axis, it records all three axes onto the SD chip for one minute using the 2 g sensitive mode. As far as using it as a seismograph, it will not be as sensitive as the poor man's version (**Figure 3**) but will work for slightly larger earthquakes, showing the direction of the waves and their g force. The earthquake is determined using the zero crossing mode which makes it very sensitive.

For a continuous monitor, all three axes are watched on any of the settings of 2 g, 4 g, or 8 g. This is also set using an Excel spreadsheet. It will continue monitoring until the program button is pressed.

The continuous monitor has many applications, e.g., racing cars, model airplanes, rocketry, etc. The SD card size will determine how long it will record; it records 50 samples per second. How about a physics project of building a container that contains an egg and dropping it from a height of 20 feet without breaking it? This device

STEPS TO USE THE WINDAQ WAVEFORM BROWSER

Getting rid of the header:
1. Insert the SD card into your computer.
2. Open the file in Excel as described in the text.
3. Remove any header and date.
4. Remove blank cells by using "Delete Sheet Rows."
5. Save the file with a different name (as a txt file).

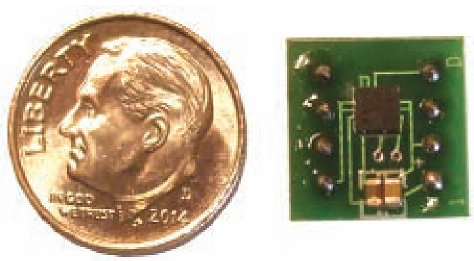
Putting your *Accelx.txt* file into WinDaq:
1. Open the WinDaq Waveform browser.
2. The screen will flash "WinDaq Waveform Browser" and then "WinDaq Playback Open."
3. Change the file type to txt mode.
4. Load the file which you just named.
5. Click on Open. The text file will need to be converted to a wdc file, e.g., *Seis1.wdc.*
6. Use a spreadsheet file.
7. Rename and save. A new black screen will open. In the center area, it will probably display "Translating 3 Channels" of data.
8. Press return and the graph will pop up.

Using the Graph:
1. Click on Options, and then click on Cursor Time.
2. At the bottom of the screen, the time and date the data was collected will display.
3. If you place the cursor under the black line and below the last red line, you will be able to read the number of seconds since the start.
4. Use the keyboard arrow keys to see the rest of the chart.

will tell you the forces involved. With drones all the rage now, how many gs can they pull?

The trip mode allows the user to set the tripping threshold from 0 g to 15 g. The data collection rate — along with amount of time recorded — is programmable.

The sensors measure 3 x 3 mm. The lower g sensor cost is minimal — around $3. They are quite difficult to solder, however, and there is no use in ruining the larger board which holds the microprocessor and SD card. I mounted the sensor on an eight-dip header (**Figure 4**).



■ **FIGURE 4.**

If you wonder why the chips are so inexpensive, they are mass-produced for handheld computers and detect if they are shaken, jarred, and rotated or tapped just by utilizing g forces. They use two capacitors with a plate in between and measure the change of capacitance between them.

The LIS3DSH has an analog to 16-bit A/D (analog-to-digital) converter and transmits the data using the I^2C protocol. By using the two's complement, it gives positive or negative g forces. The first 15 bits give 2^{15} = 32,768 bits of information. The 16th bit changes the sign, and the following data bits are another 32,768 bits of information.

The unit has a built-in clock which is programmed with the SD card. There is a power switch and pushbutton switch that program the unit. A bi-colored LED acts as an indicator for programming.

When the g force exceeds a predetermined amount, it starts the recording and indicates by a flashing LED that data has been collected. The unit draws 1 mA when in sleep mode. Two AAA batteries will provide power for

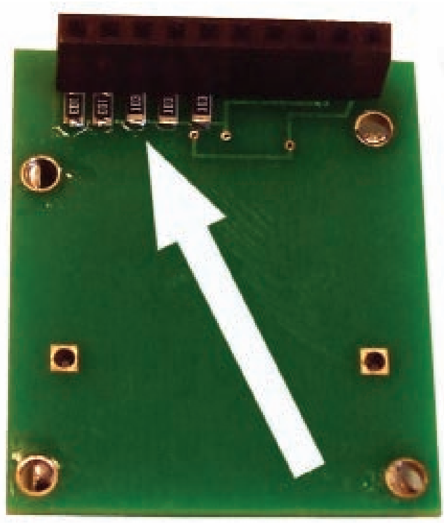
PARTS LIST

ITEM	QTY	DESCRIPTION	SOURCE/MODEL #	NOTES
Battery Holder	1 ea	Two AAA batteries		
C2	1 ea	10 µF 16V		
C3	1 ea	.1 µF 100V		
C4-C5	2 ea	33 pF 100V		
Header 1	10 ea	.1 vertical male		
Header 2	10 ea	.1 vertical female		
IC2	1 ea	PIC24FJ64GA002	Microchip	
R7	1 ea	330 ohm 1/8 watt		
R6-R8	2 ea	10K 1/8 watt		
R9-R13	5 ea	10K 805		
S1	1 ea	SPST tactile red	TI1105F250Q	E-switch
S2	1 ea	SPST slide switch	EG1903	E-Switch
SD Holder	1 ea	SD-I/O memory card holder	0678408001	Molex
T1	1 ea	Bi-colored LED		
X1	1 ea	32.768 kHz 6PF CYL		
2-16 g LIS3DSH accelerometer mounted with capacitors			*Nuts & Volts* Webstore	
100-400 g H3LIX331DLTR accelerometer mounted with capacitors			*Nuts & Volts* Webstore	

about 40 days. For a lighter weight version, a lithium coin cell can be used, which should last about 10 days. After recording, the SD card is removed and downloaded to a regular PC.

Ever wonder how hard the post office or UPS throws your package marked fragile? By changing the accelerometer chip to the ST 100-400 g, this device will show you. Its tripping point can be set from 1 g to 400 g on all three axes. It normally records for five seconds and then goes to sleep. The recording time can be changed in the Excel spreadsheet. The advantage this unit has over static g monitors is that this one dates and times the events.

For example, a three foot drop on a hard surface can produce up to 40 gs. It will date and time the event happened when its tripping point is exceeded, but will not show the amount of gs since it does not have FIFO capabilities. If you need this feature, you can run the unit in continuous mode.



■ **FIGURE 5.**

Electronics

The microprocessor used is a Microchip 16-bit PIC24J64GA200. Programming pins are made available on the board for use with PICkit 2 or 3 programmers. The software is free from Microchip. To change the programming, you need to know how to program using C. There is plenty of documentation on how to interface this chip with SD cards. (Refer to the October 2010 "Implementing a File I/O for the 16-bit Micro Experimenter" by Thomas Kibalo as one example.) If you don't have a programmer, a pre-programmed chip is available at the *NV* webstore.

The power for this chip is three volts; the accelerometer chip has a range of 2.4-3.6 volts.

If you want to make your own boards, the Express PCB files are at the article link. Express PCB provides free software at **www.expresspcb.com**.

The SD card sits above the microprocessor and is easy to remove. The board is round to fit in a 2.5" PVC EMT tube.

The microprocessor talks to the accelerometer using I^2C. The accelerometer has a built-in A/D converter and outputs the results in digital form. The micro is put to sleep in the seismic and shipping modes, thus it draws very little power.

The SD card programs the micro with the date, time, mode, and tripping point. There is an Excel program at the article link to program the SD card. Each time an event happens, the date, time, and results, are recorded on the SD card. The files are named Accel1.txt, Accel2.txt, etc.

Each event advances the name of the file up to 1,000 names. The LED will flash every four seconds indicating that an event has taken place.

The card can be removed any time by turning off the power and placing it into a computer for downloading and viewing.

Constructing the Board

There are three boards used for this project: a square board, a round board, and the accelerometer board. (The accelerometer board comes assembled in a kit, also available through the *NV* store.)

Turn the square board over and solder the five 10K 805 surface-mount pull-up resistors. The best way of doing this is to melt a small amount of solder to the top six pads. Using tweezers to hold the 805s, touch the tip of the soldering iron to the pad and place the 805 on the pad allowing the solder to flow to the resistor. Do this for the rest of the resistors.

Now, solder the other side of the resistor to the pads (**Figure 5**). Make sure you use rosin core solder; I normally use a .6 mm diameter. Turn the board over and solder the SD holder to the top side. Solder each of the pins and ears in their respective holes. Push the 10 female headers from the bottom side of the board and solder.

The round board will need to be trimmed to the outside circle. The easiest way is to use a sander. All the components are placed on the top of the board.

Solder IC2 to the board. Note the square hole is pin 1 of the chip. Place the chip in front of you with its notch pointing left. Pin 1 is the lower left pin. Usually, there is a small dimple above it. There will be a number of areas vacant, so don't panic. These were for the previous seismograph article, and are not included in the Parts List.

Solder resistors R3-R6, R8, R14, and R15; capacitors C2, C3, C4, and C5; and the crystal. Note the polarity of the 10 µF cap. Solder the two switches, then the LED with its long lead going to the square pad.

Push the 10 male headers into the top of the board where the in-line 10 pads are and solder. There will be extra pads next to the microprocessor that can be used for other projects

Pass the two wires of the battery holder through the strain relief hole and solder them to their pads. If you are

HINTS & TIPS FOR THE HOCKEY PUCK ACCELEROMETER

To view and change the board files and schematic, go to **www.expresspcb.com** and download their free CAD software. There is no obligation and you will not be harassed by emails.

Microchip provides free software for programming their microprocessors at **www.microchip.com**.

MISSING PARTS LIST

The following parts are not needed:
R1, R2, R4, C1, IC1

These were used with the "Poor Man's Seismometer" published in the *Nuts & Volts* May 2012 edition.

DOs AND DON'Ts

If data won't record on the SD card, check to see (with Notepad) if *Accel1* is corrupt. If it is, reformat the SD card and add *time.txt*.

If no file recorded on the manual setting, make sure you stop it by pushing the button. If you just turn off the power, the file will not close.

If the green LED doesn't flash, make sure the SD card is in place and locked.

When using a high g impact, I recommend using solder tab batteries. When mounted in the battery compartment, they tend to break contact upon impact and reset the machine. The red LED will turn on if this happens.

Although both chips have the ability to collect data up to 1,600 Hz, do not exceed 50 Hz as writing to the SD card along with the micro won't keep up.

When using continuous mode, be aware the unit pulls 25-30 mA Use a hefty battery for long runs.

I use the new Notepad++ for viewing raw data. It is free software.

USING THE WINDAQ WAVEFORM BROWER:

To view the data, go to **www.dataq.com** and download their free software titled WinDaq Waveform Browser (**Figure A**) under downloads. For those that are interested in great data loggers and data acquisition, I can't say enough good things about WinDaq. The browser is also located at the article link.

WARNING: YOU MAY HAVE TO REMOVE THE HEADER INFORMATION FOR THE FILE USING NOTEPAD.

Once the Waveform Browser is downloaded, there should be an icon on your computer; click this icon. A window should open up. Locate the SD card and make sure you have Files on the *.txt type. Look for the *Accel1.txt* file or other numbered seismic files.Click on this. Save *Accel1.wdc* (default). Use the #15 spreadsheet print file (ASCII) with fixed scale. Add an "a" after the file name and save. The convert screen should pop up with "Translating 3 Channels." Put in 50 for the sample rate per channel and the default on volts. Hit the return key. (This is subject to change depending on what version you have of the browser.)

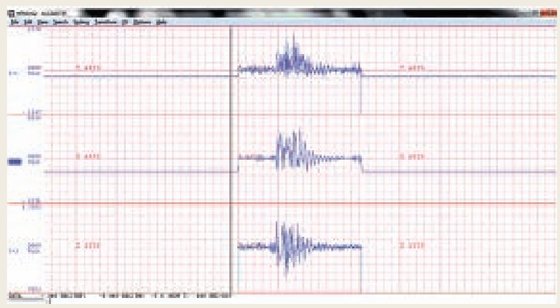
The Waveform Browser should now be visible. If you go to Options and click on Cursor Time, the date and time will show at the bottom when an earthquake happened. Moving the cursor below the red line will show the time. By pressing F4 to place a time marker, you can measure from the time marker the seconds from an event.

You can increase the amplification by centering the wave form (Z axis), changing the voltage:

1. Place the cursor on 3=3 and right-click.
2. Hold down the right button and make a box around the first data.
3. Drag the box so that the data is approximately at the center line.
4. Put the curser back on 3=3 and right-click; this will reduce the spread voltages.

There is a plethora of commands you can use in the WinDaq browser, so I defer you to their help menu as it would take a small book to explain them all. Once you have downloaded your data, delete all the *Accel.txt* files on the SD card to re-record.

Have questions? Write or call me.
Ron Newton
1035 Haystack Dr.
Carson City, NV 89705
775 560-8842
sjnewt@att.net



■ **FIGURE A.**

going to use the accelerometer as a seismic detector, I recommend you use D batteries so you won't have to change them so often. D batteries should last about a year.

The chip can be programmed or re-programmed repeatedly using a Microchip PICkit 2. The square pad is pin 1 for the programmer. There are only five pins needed (the sixth is redundant).

Container Construction

The container is made out of a 1.4" piece of 2.5" EMT PVC pipe. Both the top and the bottom of the pipe should be smooth.

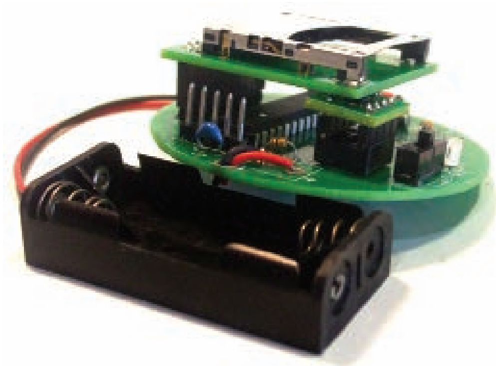
You will find a template for the Plexiglas lens at the article link. Cut it out and glue it to a 1/4" piece of Plexiglas. Sand the Plexiglas to the edge of the circle.

Using a 9/64 drill, make three holes. Remove the template using hot water. Place the lens over the smoothed end of the PVC pipe and mark the pipe where the holes are. Using a #36 drill, make the three holes 1/2" deep. Tap each hole with a 6-32 tap.

Using three 1" screws, add a drop of "super glue" to each hole and drive each screw into the PVC pipe. Use a cut-off tool and remove the heads. Glue a second piece of Plexiglas on the bottom. The top lens is secured using three knurled nuts.

Cut a piece of .06" x .125" Styrene plastic strip to 7.75 inches. Bend the strip and place it 3/4" from the top edge of the PVC pipe. Again, using super glue, attach the strip in place. This will act as a support for the board.

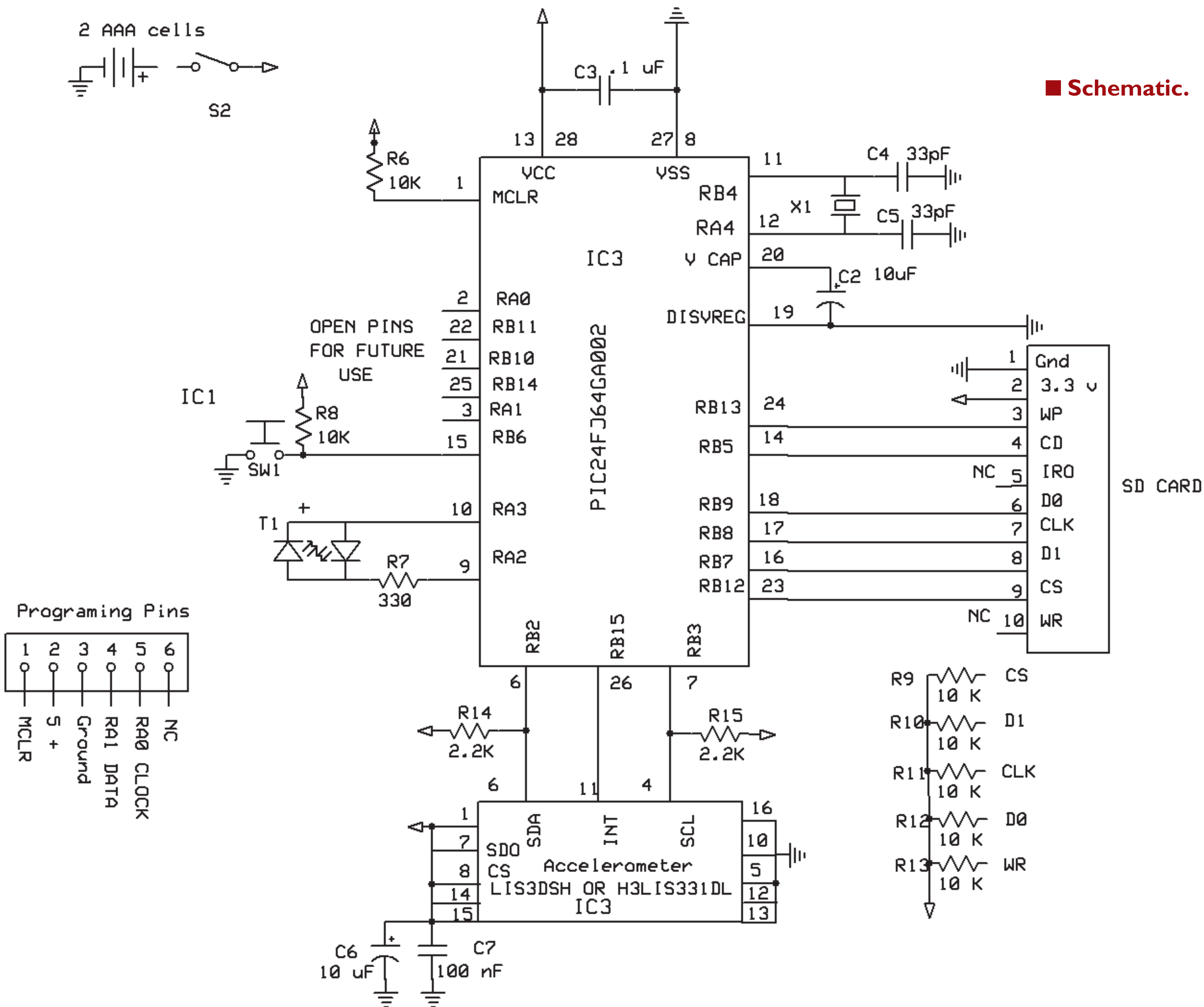
If you don't want to build the holder, the available kit comes with a tin container which the battery holder and the PCBs fit into. Just add your own foam insulation.



■ **FIGURE 6.**

Using the Unit

Plug in the accelerometer that you are going to use. Plug in the SD card holder to the 10 pins. It will rest on the accelerometer board (**Figure 6**).



■ **Schematic.**

The modes and several parameters of the accelerometer are set by programming the SD card using an Excel spreadsheet. Make sure you have the right spreadsheet. There are two: one for 2-16 gs and the other for 100 gs-400 gs.

Using your computer, format an SD card from 256 MB up to 2.0 GB using Fat 32. We are now going to set the time in the microprocessor's clock. (**NOTE: If the power is turned off, the micro's clock will be lost.**)

Download the Excel sheet titled "Accelerometer Programming 2g-16g" or "Accelerometer Programming 100g-400g." Follow the directions in the spreadsheet. The time is written to the SD card using a BCD format in hexadecimal code. Ten six-bytes of code in one long statement are stored in *time.txt*. Make sure you pre-program the time ahead to allow enough time to place the card into the unit.

Seismic Detector

In the Excel spreadsheet, type in an x next to Seismograph. Load the card and turn on the unit. The LED should be red. Watch the clock, and when it reaches the proper minute, press switch 1 and release. The LED should flash green, indicating that the micro's clock has been set to the time you programmed on the SD card.

Once you have located the area where you are going to place the seismograph, you can activate it by pushing the switch. The LED will turn on steady red for one minute. This will allow you to put the lens on and secure it with the knurled nuts. When the red LED goes out, it indicates that it is armed.

When the unit is jarred or an earthquake happens, the data will be written to the SD card and the LED will turn green, indicating that it is making a measurement. Once the unit has made a one minute measurement, the LED will flash red about every four seconds, indicating that an earthquake (or event) has occurred.

When you are ready to read the data, remove the lens (this will activate the unit and record the jarring from the removal, but will also end up as an invalid *txt* file). Turn the power off and remove the SD chip.

Continuous Monitoring

In the Excel spreadsheet, type in an x next to Continuous Monitor and load the card. Turn on the unit;

the LED should be red. Watch the clock, and when it reaches the proper minute, press switch 1 and release. The LED should flash green, indicating the micro's clock has been set to the time you programmed on the SD card.

Press the switch; the LED should flash red and go dark, indicating that it is collecting data. I turned the LED off to save battery power. To stop recording, press the switch again. The LED should turn red. Turn the power switch off and remove the SD card. The g ranges can be changed to 2 g, 4 g, 8 g, or 16 g using the spreadsheet. Note: When the unit is used for continuous monitoring, the SRAM card becomes a power hog pulling 25 mA constantly, whereas the seismic mode or tripping mode only pulls 1 mA.

Tripping Monitor

In the Excel file, type in an x next to Tripping Monitor. You'll need to type in the trip point you want the unit to start recording at. This can be 0 .1-15 g. Follow the instructions for loading the Notebook file in the spreadsheet.

Load the card and turn on the unit. The LED should be red. Watch the clock, and when it reaches the proper minute, press the pushbutton switch and release. The LED should flash green, indicating the micro's clock has been set to the time you programmed on the SD card.

Once you have decided where you are going to place the shipping monitor, you can activate it by pushing the button. The LED will turn on steady red for one minute. This will allow you to put the lens on and secure it with the knurled nuts. When the red LED goes out, it indicates that it is armed.

When the unit is dropped, jarred, or thrown, and exceeds its trip point, it will record for five seconds (this can be changed) when an event happens. The data will then be written to the SD card. Once the unit has made a measurement, the LED will flash red about every four seconds indicating the trip point has been exceeded. NOTE: Only one mode can be used at a time.

Viewing the Data

Using Excel for Downloading Data

Excel can be used for short runs, but can only hold 1,048,576 readings and will only graph 32K. The Dataq Waveform can hold a lot more. However, the Excel is easy for reading one minute seismic data.

1. Open Excel.
2. Click on the Office Button (very upper left corner) and then Open.
3. Change the All Excel Files to All Files in the bottom right corner.
4. Locate the SD card and click on one of the *Accelx.txt* files.
5. Make sure the Delimited box is checked, then click Finish.
6. The file will load.
7. Put the cursor on A8 and hold down the shift and Ctrl; use the down arrow and it will highlight column A. Press the right arrow button and it will highlight all the cells. You don't have to use the whole file.
8. Click on Insert, then Line, and the top middle chart. The chart will display the data.

I hope you'll be able to shake things up with this unit. Happy egg dropping! **NV**

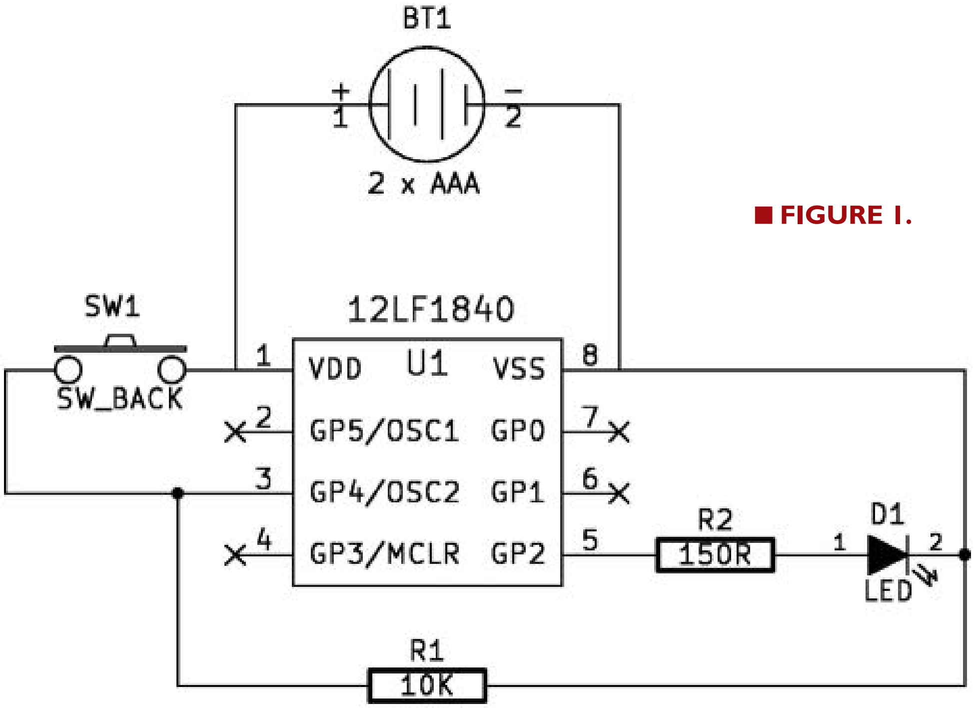
Use a Pushbutton to Turn Your Devices On and Off

TOSS THE TOGGLE

Most electronic devices today have a single button you push to turn them on and off. Think of your cell phone, laptop, and even your TV. There is no toggle to flip, no knob to turn back and forth, nor slide switch to move. So, how do you get one of these power buttons into your project so your latest gadget can sit next to your other devices without the embarrassment of a toggle or slide switch?

Microprocessors drive most of our devices, and many of these processors have the ability to go into a low power sleep mode. In this state, the processor can still respond to an external signal such as a button press. When you combine that with some circuitry to control power to peripheral devices such as displays, radios, and servos, you have a system that can turn itself on and off without much additional hardware. The December 2013 issue of *Nuts & Volts* described an add-on solid-state power switch from Pololu which performs a similar function. As a regular customer of Pololu, I have been aware of this module for a while. However, I wanted a cheaper, more flexible, and integrated solution that wouldn't require me to stock another part. If you're using a microprocessor and you're in control of the hardware and software design of your system, you can get this power control capability at little to no extra cost in money or board space.

By Jürgen G. Schmidt



■ FIGURE 1.

I've been working with Microchip PIC processors for quite a while, and their eXtreme Low Power (XLP) processors have some impressive low power sleep modes — with some claiming 20 nano-amps (nA). This makes them ideal for battery-powered devices. Okay, but what does that mean in real life?

We need to do a little math to get some perspective. Let's say the design goal is to run the device off two AAA alkaline batteries for at least a year. The actual run time of a wireless doorbell transmitter, for example, is calculated as 1,000 rings per year, with each ring lasting a generous four seconds and drawing 25 mA. That comes to a total of roughly 28 milli-amp hours (mAh). The total capacity of these batteries ranges from 800 to 1,200 mAh, so we can easily operate the doorbell for several years.

Now, we need to worry about the current draw while the doorbell is idle. If our circuit at idle draws 1 micro amp (uA) or less, then the batteries will last roughly 800,000 hours. With 8,760 hours per year, this is about 90 years. At a sleep-mode draw of 20 nA, we're talking about 50 times longer!

So, if the specs are correct, then putting our processor and the rest of the device into sleep mode should not appreciably deplete our batteries. The shelf life of alkaline batteries is only 10 years, and most gadgets don't stick around for even that long — with garage door openers and alarm sensors being the exception. For now, on paper, the numbers look promising, and if the spec sheet writer hasn't misplaced a decimal point, we can use a low power microprocessor to switch our device on and off.

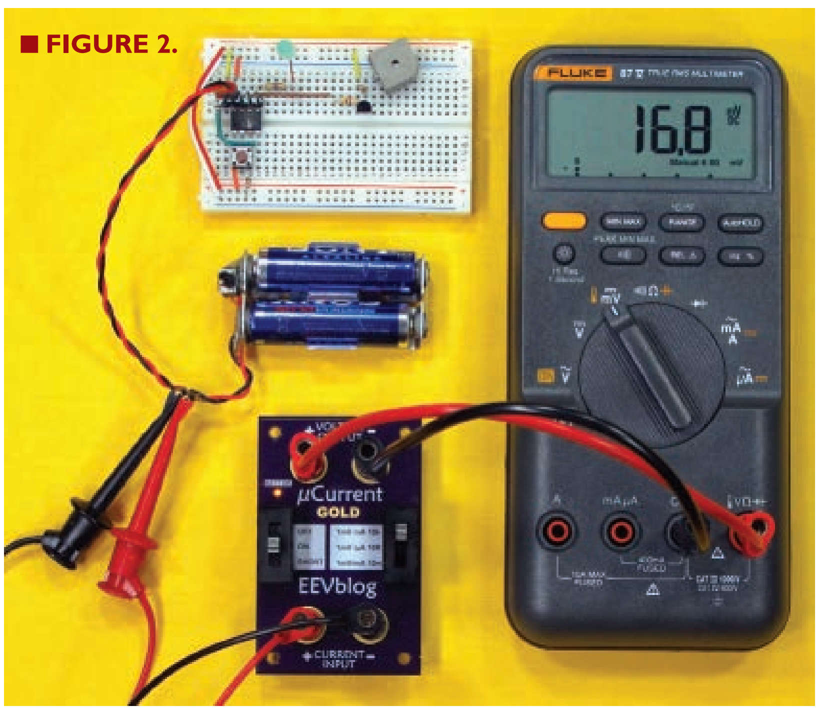
Design and Testing

A simple pushbutton operated LED is used to test this. I only needed a few parts (which I had on hand), a voltmeter, and David Jones' µCurrent (see sidebar) to verify I'm drawing very little current when I've turned the system off. **Figure 1** shows the test circuit. You can see that the processor is permanently connected to the battery. The power switch is between the battery and one of the processor input pins.

Figure 2 shows the prototyped circuit (with a few extras) connected to my current measuring system.

The meter shows that in sleep mode this circuit draws 16.8 nA. The code for the power switch is written using the CCS PIC® C compiler, and the entire file can be obtained from the article link. Portions are duplicated here to explain how the switch works.

The power switch uses two key features of the



■ FIGURE 2.

Post comments on this article and find any associated files and/or downloads at **www.nutsvolts.com/index.php?/magazine/article/november2014_Schmidt.**

References

Vendor Websites:

CCS PIC® C Compiler
www.ccsinfo.com

OSHPark PCB Manufacturing
oshpark.com

Mouser Electronics
www.mouser.com

Microchip
www.microchip.com

Documentation:

David Jones Video Blog — 600+ videos!
www.eevblog.com

Microchip *Tips & Tricks Guide;* Section 2 addresses low power issues
ww1.microchip.com/downloads/en/DeviceDoc/01146B.pdf

Microchip *Low Power Design Guide* is a bit more technical
ww1.microchip.com/downloads/en/AppNotes/01416a.pdf

David Jones' µCurrent current adapter for multimeters
www.eevblog.com/projects/ucurrent

µCurrent Gold Kickstarter Project
www.kickstarter.com/projects/eevblog/current-gold-precision-multimeter-current-adapter

Adafruit µCurrent listing — an excellent short description
www.adafruit.com/products/882

processor: 1) the low power sleep mode; and 2) a "wake on interrupt" which can come from a voltage change on any of the I/O pins. These features can be found on microprocessors from many different manufacturers (not just those from Microchip). Starting from sleep mode, the general operation is as follows:

1. The user presses the power button.
2. The processor detects the change from low to high on a pin and wakes up.
3. The processor goes on to initialize itself and continues to perform its main task which, in this case, is to turn on an LED.
4. The main processing loop checks for a button press of a certain length of time (two seconds, in this case). This can be with the same button that turned the system on.
5. Once that button press has been detected, the processor turns off all peripherals, sets up the conditions to detect the next "power on" event, and goes to sleep.

Let's look a little closer at how this happens.

The system doesn't start out in sleep mode. When it first gets turned on (for example, when we put in new batteries), we don't want it to interpret this as a start signal. Fortunately, the processor can determine how it got started.

One of the first commands executed after startup is:

```
if( restart_cause() == NORMAL_POWER_UP )
HWSleep();
```

This essentially asks, "What woke me up?"

There can be several reasons for a processor waking up; the two we are concerned with are *NORMAL_POWER_UP* and *RESET_INSTRUCTION*. When we connect the battery for the first time or replace the batteries, the processor wakes up and detects a *NORMAL_POWER_UP* as the reason. The code then tells it to go to sleep since we only want it to turn on after a reset that was initiated by a button press.

The function *HWSleep()* prepares the processor for the lowest possible power consumption, sets up the interrupt, and puts the processor to sleep:

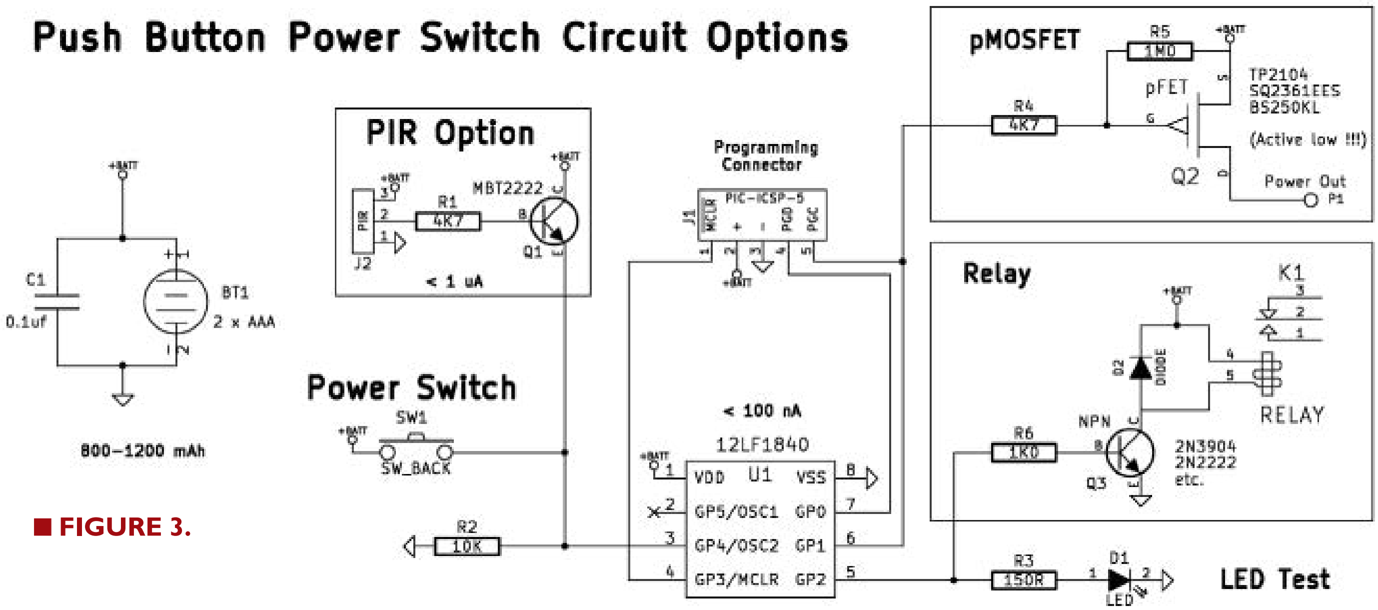
```
void HWSleep( void )
{
    //— power down what we can...
    setup_adc_ports( NO_ANALOGS );
    setup_vref( VREF_OFF );  // uses 16uA when on
    setup_timer_1( T1_DISABLED );
      // turn off timers
    setup_timer_2( T2_DISABLED );
    output_a( 0x00 );      // turn off everything
    output_high( PFET );
      // set high to turn off pFET
    enable_interrupts( INT_RA4_L2H );
      // turn on interrupt on power button
    sleep();               // go to sleep

    // Wakeup from sleep resumes here...
    clear_interrupt( INT_RA4_L2H );
      // clean up interrupt from sleep
    disable_interrupts( INT_RA4_L2H );
    reset_cpu();
      // do a full reset when we wake up
}
```

All the instructions up to *enable_interrupts()* are designed to eliminate any current leakage through the processor, and to turn off power to peripheral devices. The specific instructions will depend on the internal and external peripherals, special features of the processor, and so on. The spec sheet has all the information you need. However, you may need help from user forums and other documentation to make sense of it, and some of it is just trial and error.

For example, I had a serial connection I was using for debugging, and I just couldn't get the current draw near to what I was expecting. Once I disconnected the serial cable, all was fine. So, for those applications that have a serial connection built in, you need to add additional commands to be sure it doesn't draw power from the processor during sleep mode.

The *enable_interrupts()* instruction tells the system which pin will be used to wake it up, and the next instruction puts it to sleep. When the selected pin detects a change (in this case, a transition from low to high), the processor wakes up and resumes processing from the point at which it went to sleep.

Then, we clean up our interrupt handling and force a restart of the processor, so it can start fresh by going



■ **FIGURE 3.**

through a complete system setup.

When it encounters the check for why we started up, the reason will be *RESET_INSTRUCTION*, and processing will continue to turn on the LED and then into the infinite loop that is the basis for most embedded systems. Within that loop, we check for a button press of two seconds or longer which initiates the *HWSleep()* function all over again.

The additional hardware on the prototyping board represents control of power to a load (in this case, a buzzer) via a pFET (P-channel MOSFET). Support for this is included in the code file (*LF18Switch.c*) at the article link.

Applications

Using a microprocessor as a power switch may seem like over-kill, however (as I mentioned earlier), since my projects are usually processor-based, I get the functionality for free. Even the pin for the button is free because I can use it for other operations after the system is turned on. While some systems — such as our LED light — need a switch to turn off, others will turn themselves off automatically after they have completed their tasks. We'll see that in my wireless doorbell and motion sensor transmitters next.

Before we get to that, there are some other tests I want to perform. Turning an LED on and off is a great proof of concept, but I need to do some real work. In **Figure 3**, I show a relay and some other peripherals added to the original circuit. A transistor-driven relay allows us to control large loads. Beware that when the system is on, the relay represents a significant load, and unless it is turned on only briefly, is not suitable for a small battery-powered system. No change is required in the software to use a relay instead of an LED.

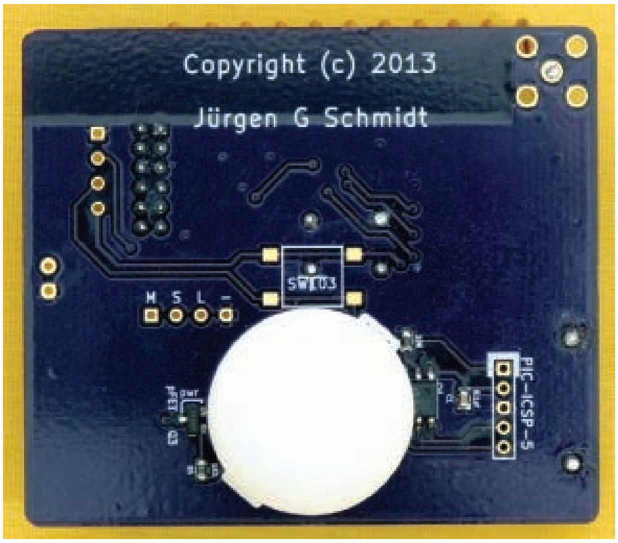
For driving relatively low power (200 mA or less) loads, I prefer to use a pFET. They're cheaper than a relay, use very little space, and can have a very low voltage drop. Driving a 20 mA radio transceiver, I experienced a "diode forward voltage drop" of only 0.15V. Ordinary transistors will usually have a drop of 0.7V or more. The 2N2222 I tested had a 0.9V drop with the same 20 mA load I tested with the pFET.

For my application, I need to get the highest possible voltage from the battery to my radio. The circuit for this is included in **Figures 2** and **3**. Note that the pFET is active low. You need a 1M pullup resistor (R5) to ensure it turns off because the output from the processor will float when it's asleep.

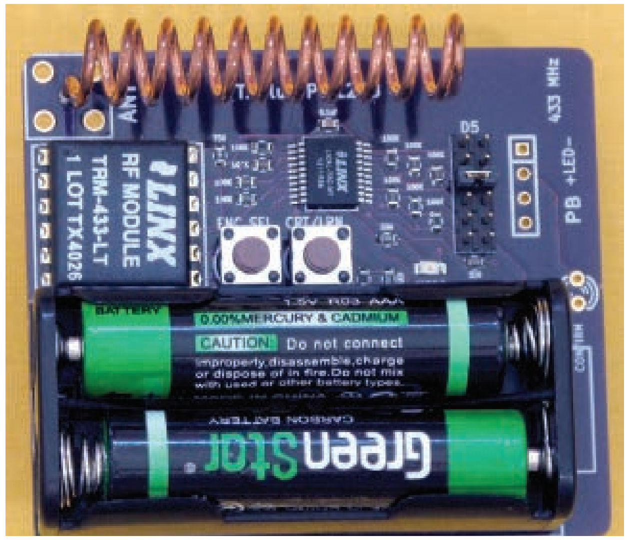
The reason I'm focused on low sleep currents and pushbutton activation is a wireless doorbell and alarm project I'm working on. I want to be able to use a pushbutton switch or a PIR (passive infrared) sensor to trigger the radio transceiver that communicates with the base station. The radio takes a while to power up, so if you just connect the pushbutton between the battery and radio

DESIGNATOR	PART	CATALOG PART #
(All part numbers are for Mouser Electronics.)		
U1	PIC12LF1840	579-PIC12LF1840-I/P
Q2	pFET TP2104	689-TP2104N3-G
PIR	Panasonic	769-EKMB1103111
The other parts are miscellaneous workbench supplies.		

■ FIGURE 4.



■ FIGURE 5.



Measuring Microamps and Nanoamps

In the course of working on this project, I discovered that measuring current in the micro-amp and nano-amp (nA) range is no easy task. Most digital voltmeters can measure milliamps. However, measuring anything below that with accuracy is difficult. Even my multi-hundred dollar professional multimeter was not up to the task. To verify that my circuit was truly drawing a microamp or less, I needed a reliable measuring device.

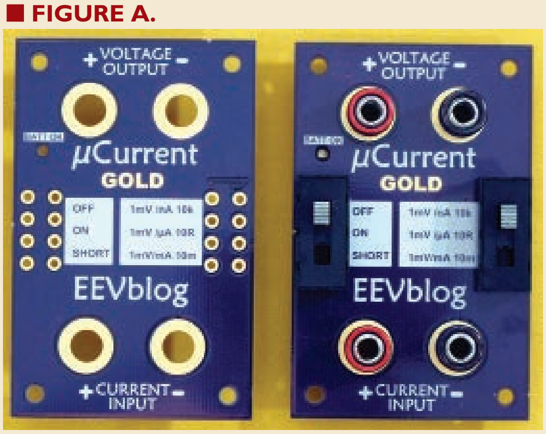
In researching for a solution to this, I discovered a device made by David Jones in Australia. If you have been learning about electronics for a while, you may have come across one of David Jones' enthusiastic videos on YouTube or his video blog. He has over 600 videos covering basic electronics, how to use your oscilloscope, product reviews, and more. I've learned a lot from him over the years.

In 2009, David Jones presented the open source µCurrent, and described both the need for it and how to make it in an article in the April 2009 issue of *Silicon Chip Magazine*. The µCurrent is a low cost (less than $100) adapter that lets you accurately measure current down to the nano-amp with any digital multimeter capable of measuring millivolts. In 2013, Jones announced the version 5 µCurrent Gold and initiated a Kickstarter project to fund production, exceeding his funding goal more than ten-fold. His video on the Kickstarter page does a good job of explaining why you need something like the µCurrent to measure current without impacting your circuit.

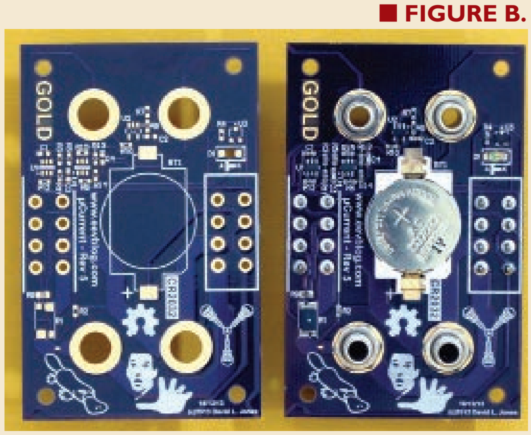
He is currently in production, but it may still be a while before units are available in the US. I wasn't prepared to wait, so I downloaded the publicly available PCB manufacturing files and parts list. I made some changes to the PCB files so they would be accepted by OSHPark, and I checked with David Jones about substituting for a part that was only available in full reel quantities. All of the parts were available from Mouser Electronics. Once the boards came back, I assembled my own µCurrent and sent David Jones a donation. You can see the bare boards and my assembled µCurrent in **Figures A** and **B**. The µCurrent uses several precision op-amps and precision resistors to generate a voltage in the millivolt range that is proportional to the current being measured. The µCurrent can be seen in action back in **Figure 2**. The 16.8 millivolts displayed on the meter translates to a measurement of 16.8 nA for this circuit in sleep mode. I'm using a PIC12LF1840 microprocessor, and the spec sheet says the sleep mode current is 20 nA. For anyone working with low currents, a tool like the µCurrent is indispensable.

See the **References** section for more information about David Jones, the µCurrent, and the vendors I use.

■ FIGURE A.



■ FIGURE B.



module, and then tap the button, the circuit may not be on long enough to power-up and send the signal. I need to be sure the transceiver has enough time to fully power-up, send a signal, then continue to stay on long enough to receive a confirmation signal. This, in turn, lights the doorbell button to let the user know the bell has actually been rung.

The processor provides the necessary logic and timing. In this particular scenario, the pushbutton is only used to turn the system on. Once the processor has performed its tasks, it goes to sleep on its own, waiting for the next visitor.

The entire package fits into a plastic single-gang junction box behind the doorbell button. When idle, the circuit draws less than 50 nA. On average, the system is powered to 20 mA once a day for only a few seconds. The batteries will easily last more than a year, which was the design target.

Figure 3 includes a PIR motion sensor trigger circuit for the same wireless doorbell project. I have several transmitters, and the receiving base station plays a different tone depending on which transmitter is triggered. Panasonic makes a low power PIR motion sensor that draws less than 1 uA while waiting to detect motion. When motion is detected, the signal is amplified by an NPN switching transistor that, in turn, triggers the interrupt on the processor, waking it up.

In addition to driving the radio with the necessary timing, the processor has some delays built into it so the receiver isn't constantly ringing while something is passing the motion sensor. The radio is turned off during these delays. **Figures 4** and **5** show the front and back of the wireless motion sensor module.

The front has the radio, antenna, and battery pack. The back has the PIR sensor, power control, and processor. This is designed to be mounted discretely in a junction box with only the sensor dome exposed. I used OSHPark (see **References**) to manufacture the circuit board. Assembly is done via my toaster oven reflow system.

I hope this has inspired you to "toss the toggle" and include a pushbutton power control into your next project. While I have focused on a particular microcontroller and compiler, these principles can be applied to other hardware and software development systems. Email me with questions and/or comments at jurgen@jgscraft.com. **NV**

Tools of the Trade

Most people who know me, know that I am a "tools over toys" kind of guy. That is, I would prefer to have a new useful piece of test equipment over some new tech toy. You may have heard the quote, "The right tool for the job." In any profession (or project build, for that matter), things go smoother when you have the correct equipment to accomplish what you're trying to do.

By Chris Savage

Post comments on this article at **www.nutsvolts.com/index.php?/magazine/article/november2014_Savage**.

History

When I first got into electronics and computers, I spent all my money on what I will call "toys." Some new IC or computer peripheral would come out and I had to have it. This resulted in no budget for "tools" — especially test equipment. When something needed to be repaired, I had to resort to very primitive methods to troubleshoot, borrow the tool, or, in some cases, build my own test tools.

Over the years, my philosophy has changed. It is more important to me these days to have decent tools and test equipment before I even think about buying parts or things I don't need — no matter how good of a deal it is. My packrat status has evolved into a need for useful tools to help me realize not only my project goals, but to troubleshoot, debug, and even hack.

An Efficient Workstation

It used to be that the majority of my tools were homebrew. Over the years, I have learned which commercial tools I prefer to use, and have acquired most of what I need to accomplish the kind of work I do these days. I work as an engineering tech for Parallax, Inc., and I spend most of the day at my workstation (**Figure 1**), so it's important for me to have everything I use the most available to me at my bench so I don't have to go looking for it (or worse, not have it).

FIGURE 1.

Looking at my workstation, you can see most of the tools I tend to keep handy on or near my bench. Some are in a rolling drawer cabinet to the left of my work area. In my normal duties for Parallax, I provide technical support to customers having issues getting products to work. Sometimes this involves determining if the customer has the appropriate test equipment to do things, like obtain voltage measurements on various pins, check for communication, etc.

I also design some of the sensors and other products Parallax offers. For this task, I especially need the right

tools. Often it's equipment the average customer would not typically be using, but the kind of stuff other developers might be.

I'm often asked by customers, "What tools do you use? What do you recommend?" I'd like to answer this question by exploring the tools I feel are important on any bench whether you work with electronics and/or microcontrollers as an experimenter, hobbyist, technician, or engineer. Let's start with test tools.

FIGURE 2.

Digital Multimeters

The most basic test tool you should have is a digital multimeter (DMM). I can't overstate how important this tool is. There are so many times an issue is related to something simple like low voltage, current issues, a broken trace/wire, or something else that can be easily detected with a DMM. A basic unit can help you obtain voltage and resistance, and can be found for around $10 from many retail sources. Of course, if you prefer more quality, features, and safety, you'll want to invest in a more decent unit, especially if you use it as often as I do — which is almost every day. I prefer Fluke digital multimeters as do most of my colleagues and peers. Fluke units are high quality dependable tools that offer high accuracy and robust features.

Figure 2 shows my favorite Fluke DMM models and the ones I use for personal and work projects. Both models offer quality and features not available in cheap multimeters, and a level of reliability and consistency that makes the choice easy for me. In addition to measuring AC/DC voltage/current, resistance, continuity, capacitance, and temperature, they also have a diode checker, analog bar display, and a host of other features. Be sure to go to the Fluke website for more information.

FIGURE 3.

Logic Analyzers

A logic analyzer is a must-have for the type of work I do. This device can capture digital signals, allowing you to easily detect missing signals or improper timing. I use a logic analyzer for timing and communication issues.

Most logic analyzers have protocol decoders which allow you to see the communication on various types of busses such as SPI, I^2C, asynchronous serial, and more. If you work with these signals and have a need to see what is being sent on various busses, then a logic analyzer is truly valuable.

My logic analyzer of choice is the Saleae Logic (**Figure 3**). I have been using the 16- and eight-channel models for several years. Saleae is in the process of releasing some newer logic analyzers which also support analog, however, the new hardware was not available for this article.

Unlike most of my other test equipment, the Saleae Logic does require a PC and is a USB device.

Oscilloscopes

A good oscilloscope is an important piece of equipment and — for some reason — was the hardest tool for me to acquire when I first got started. Scopes were expensive in the '80s and '90s, and I just didn't have the funds to purchase one. Older Tektronix analog scopes were how I got started. The first scope I ever owned was a Heathkit scope that was partially assembled and that I picked up in the late '90s. Later, I was able to acquire a more modern Tektronix two-channel digital oscilloscope. When I started working at Parallax, most of the scopes were Tektronix digital units; some had storage capability and other features.

An oscilloscope can help you see things that a DMM and a logic analyzer cannot. A DMM is typically too slow to show rapid changes in voltages or currents, and therefore cannot see noise, dips, or spikes. A logic analyzer only shows logic ones and zeros, so voltages that are outside of the logic threshold are often invisible to this device. A scope can see a signal in real time, showing a graphical representation of the signal on the screen. The scope is looking at a relatively narrow period of time. By expanding that to be more visible, you can see the details of the signal such as rise/fall time, level, noise, and other properties.

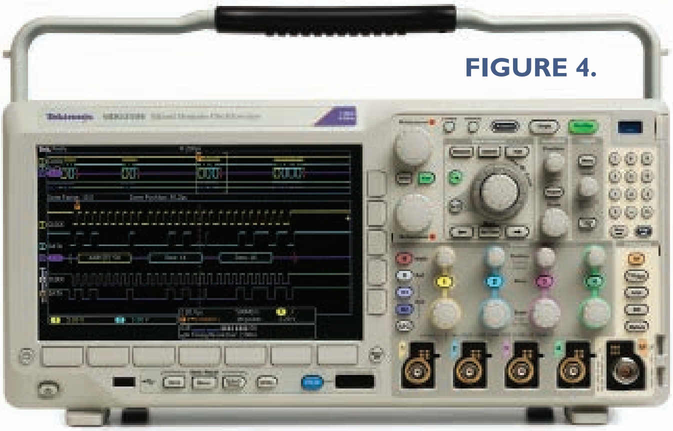


FIGURE 4.

FIGURE 5.

FIGURE 7.

Oscilloscopes also have the ability to trigger on a signal or some external event. Modern scopes have even more capabilities that let you see and troubleshoot many types of signals, as well as store and analyze data. One such scope is the MDO3000 series from Tektronix (**Figure 4**). See their website for more details on this robust unit.

The MDO3000 series is actually six instruments in one tool: an oscilloscope; spectrum analyzer; logic analyzer; arbitrary/function generator; protocol analyzer; and digital voltmeter. So, having this tool on your bench can replace several pieces of equipment.

This device is also ideal in a manufacturing or production environment for anyone who designs products that might require the tools included in this scope. I realize that the average experimenter or hobbyist cannot necessarily afford a full-featured scope like this, however Tektronix has other options.

A decent entry-level scope from Tektronix is the TBS1022 (**Figure 5**) which provides two channels and a lot of features at a very nice price. I would recommend this as a starter scope if you're looking. I have found Tektronix instruments to be solid, reliable, and accurate.

If you happen to fall somewhere in between these two scopes, I have also owned and used the TDS2024B and the MSO2024 which are both solid scopes and have a great price-point for their features (**Figure 6**).

Another oscilloscope I have not personally used but wanted to mention is the WaveSurfer 3000 from Teledyne LeCroy (**Figure 7**). The thing that caught my attention about this scope series was the interface. I'm a geek and I tend to like scopes that have a lot of knobs and buttons — especially if they're well placed and functional. However, some people find instruments with a lot of controls to be intimidating and even confusing. So, for those people, Teledyne LeCroy offers an advanced interface called MAUI. All the important oscilloscope controls are accessed through an intuitive touch screen.

Other Test Tools

Many of you probably won't recognize the tools in **Figure 8**. This is a Micronta logic probe and pulser set. Every once in awhile, I find myself working with discrete CMOS and/or TTL logic, so it is at these particular times I break them out.

These tools aren't very popular anymore and I don't use them very often, but I wanted to mention them

FIGURE 6.

FIGURE 8.

FIGURE 9A.

FIGURE 9B.

FIGURE 10.

because they're actually quite useful. The fact that someone like me can still find an opportunity to utilize them is pretty cool. Before I bought this set back in 1990, I built a set from scratch.

You probably wouldn't think of a power supply as a test tool, but it actually is. **Figure 9A** shows my main bench supply which is an Agilent E3644A; **Figure 9B** shows my backup unit which is a BK Precision 1697. Sometimes you need to produce a specific/known fixed voltage, and you can count on a good digital power supply to do that where your linear regulator, wall wart, or battery pack won't.

These supplies can also be set to limit the current available on the output which can assist in troubleshooting a circuit — especially when testing for short circuits. Notice that I mentioned digital power supplies? I've used a few analog bench supplies and have seen many issues with these, such as the potentiometer for voltage or current adjustment getting dirty and throwing the settings off. Or, the panel meter on the front not showing accurate voltage/current on the output. That's not to say this doesn't happen with digital units, of course.

Soldering is a core task in my profession and hobby, so for me a good soldering station is a must. I have a HAKKO 936 soldering station that I am very happy with (**Figure 10**). I used one at work years ago and later purchased a set of two from eBay with tip cleaners and extra sponges. I also have extra solder suckers for desoldering.

I currently don't have a desoldering station or a hot air pencil for SMD work, but wish I did. These are fast becoming necessary for the work I do and are slated for future acquisition.

Bench Tools and Holders

Besides test tools, any good bench needs hand tools and other support devices to design, build, and repair. One tool I used to use all the time for various things was a little set of helping hands I got from RadioShack many years ago. It was clunky, loose, and tipped over frequently, but it was what I had. Then, I found PanaVise.

The PanaVise Model 201 (**Figure 11**) — also known as the PV Jr. — is so popular that once I found out about it, I introduced it to Parallax. We now use them in soldering kits for expos and other events. This little guy is solid and has many options for mounting. In fact, I learned


FIGURE 11.


FIGURE 12.

early on that PanaVise products aren't very expensive. Plus, they have a large selection of vises and PCB (printed circuit board) holders, not to mention camera mounts and other accessories.

The PanaVise Model 324 Electronics Work Center (**Figure 12**) is my choice for larger PCB work, and has the solder and soldering iron holder — although I don't actually attach those accessories. I just hang the solder spool off the main rail. I actually have several PanaVise tools and one of their camera holders.

Speaking of holders, I also have Pomona and E-Z-HOOK test lead holders (**Figure 13**) hanging up to store my scope probes, power supply leads, and USB cables when I'm not using them. This is a very useful system of management because when the leads were left connected and on the bench, they would get tangled and just made a mess. This was even worse when I stored them in a drawer and would pull them out all tangled into each other. So, while these aren't "bench" tools, they do benefit your work area by helping keep it clean.

Did you ever lay out a PCB and find that the holes in the pads were too small for the leads of a part? How about the mounting holes or even the dimensions? Well, before getting myself a set of digital calipers I used to have to guess on measurements for PCB layout all the time. Digital calipers (**Figure 14**) take all the guesswork out, and you can find inexpensive sets of these at places like Fry's Electronics or MCM Electronics. I haven't found a particular name-brand set I prefer yet.

Electrical tape does not make a good protection for spliced and soldered wires — especially when you're soldering to component leads. You really need to use heat shrink tubing to get a professional clean finish. Often, heat guns can be expensive but you can get one on a budget (**Figure 15**). The unit in **Figure 16** was purchased at Fry's Electronics for about $15, leaving me enough to stock up on various sizes of heat shrink tubing.

A glue gun is one of those tools that I am always glad to have when I need to affix an LED or some other component in place in the absence of a hardware mounting option.

Figure 17 shows my two hand-tool drawers in the cabinet next to my bench. The top drawer (left) shows all my screwdrivers, while the second drawer (right) is mostly diagonal cutters and pliers of various configurations. If it seems like a lot of redundant tools, it's because I have a set a diagonal cutters for snipping tie straps and thicker wires and connectors.

Another set is just for trimming leads, while another is for clipping the leads from PCB components. I also have three sets of needle-nose pliers I use for different purposes. I even have two sets of wire strippers.

One set is fixed at 22 gauge wire, while the other is adjustable. I find having different sets for different jobs more efficient, and the wear is spread out over multiple tools. Anything I think might be replaced a lot is a brand

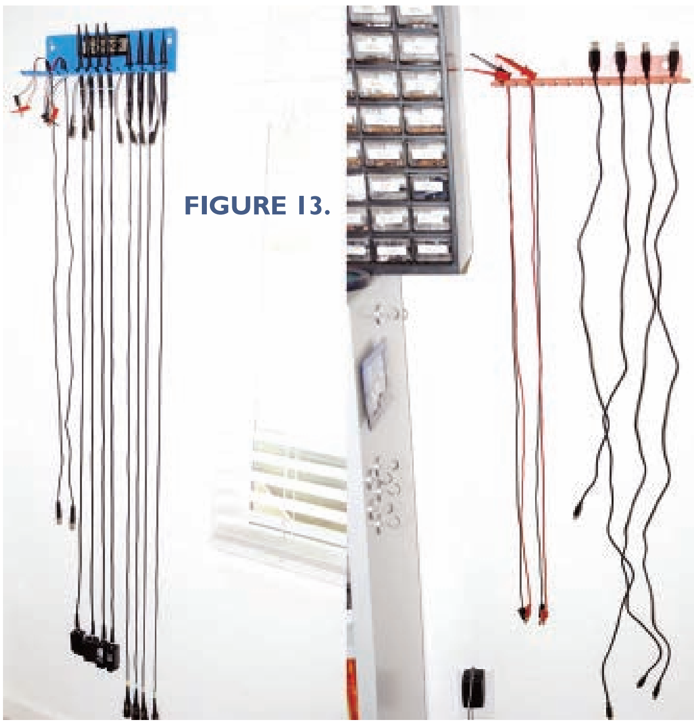


FIGURE 13.

FIGURE 16.

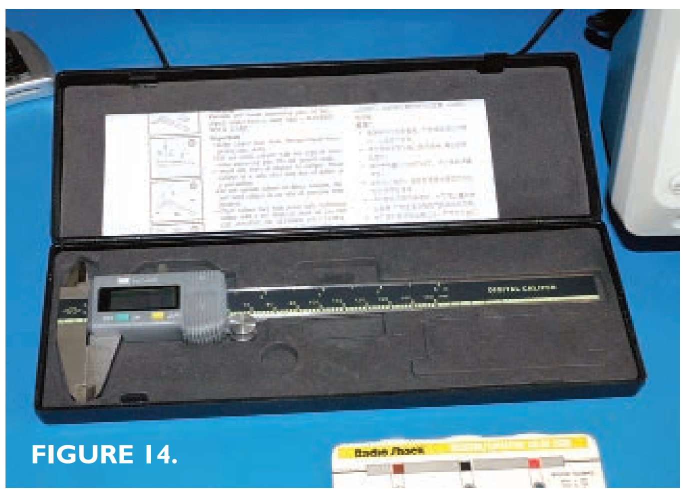


FIGURE 14.

FIGURE 15.

FIGURE 17.

that is guaranteed for life, such as Craftsman.

The last tool on my bench that I want to mention is my magnifying lamp (**Figure 18**). When working on small parts — especially SMD components — this lamp comes in very handy. I picked it up from MCM Electronics for under $50 and it is very solid. I supplement this lamp with several 10X loupes I keep handy and bought for $1.50 each.

Power Tools

I didn't mention power tools because I keep those in my garage and have a separate bench out there where I cut and drill to keep dust outside the house when possible. The two most important power tools I have are my DeWalt cordless drill and my Dremel. Both tools are essential to my work, and both are used at least weekly for some task. The cordless drill has both drill bits and taps which allow me to thread holes into metal for screws. My Dremel tool is often used to cut holes, screws, or debur and polish panels.

Always wear proper eye protection when using power tools, and even when soldering or trimming leads! Never underestimate the damage a flying component lead or molten solder could do to your eyes.

Special thanks to Amy at Tektronix and Tom at PanaVise.

Resources

Parallax, Inc.
www.parallax.com

Fluke
www.fluke.com

Saleae
www.saleae.com

Tektronix
www.tek.com

Teledyne LeCroy
http://teledynelecroy.com

PanaVise
www.panavise.com

Fry's Electronics
www.frys.com

MCM Electronics
www.mcmelectronics.com

Savage///Circuits
www.savagecircuits.com
info@savagecircuits.com

Final Thoughts

Did anyone notice the blue mat that covers my entire bench? If you haven't or didn't know what it was, it is an anti-static mat that is grounded at the corner of the bench. This provides static protection when I am working on the bench — especially since the weather here is always dry and everything is carpeted. If you don't have one of these, you could use an anti-static wrist strap. Don't underestimate the importance of static precautions when working with electronic components and devices either.

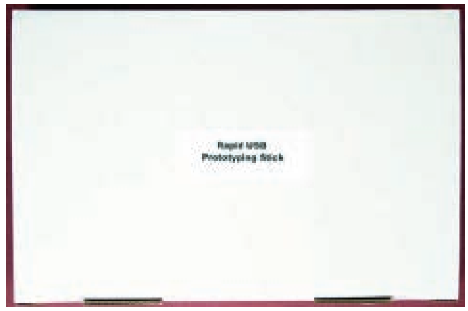
As often as I am asked what tools I use, I hope this article not only answers those questions but also offers some insight into my motivation for selecting some of the tools and brands I use.

I think of myself as pretty particular about my tools. Since I make my living using them, I do not loan them out. However, I have been known to pass tools on that I no longer use, but that are still viable. If you are the recipient of such tools, then you know how I feel when I acquire one from someone else. **NV**

FIGURE 18.

■ BY FRED EADY

PICs and a Stick Pack Big Computing Power in a Small Package



■ Photo 1. The label on the box suggests that its contents are small in stature but huge in performance.

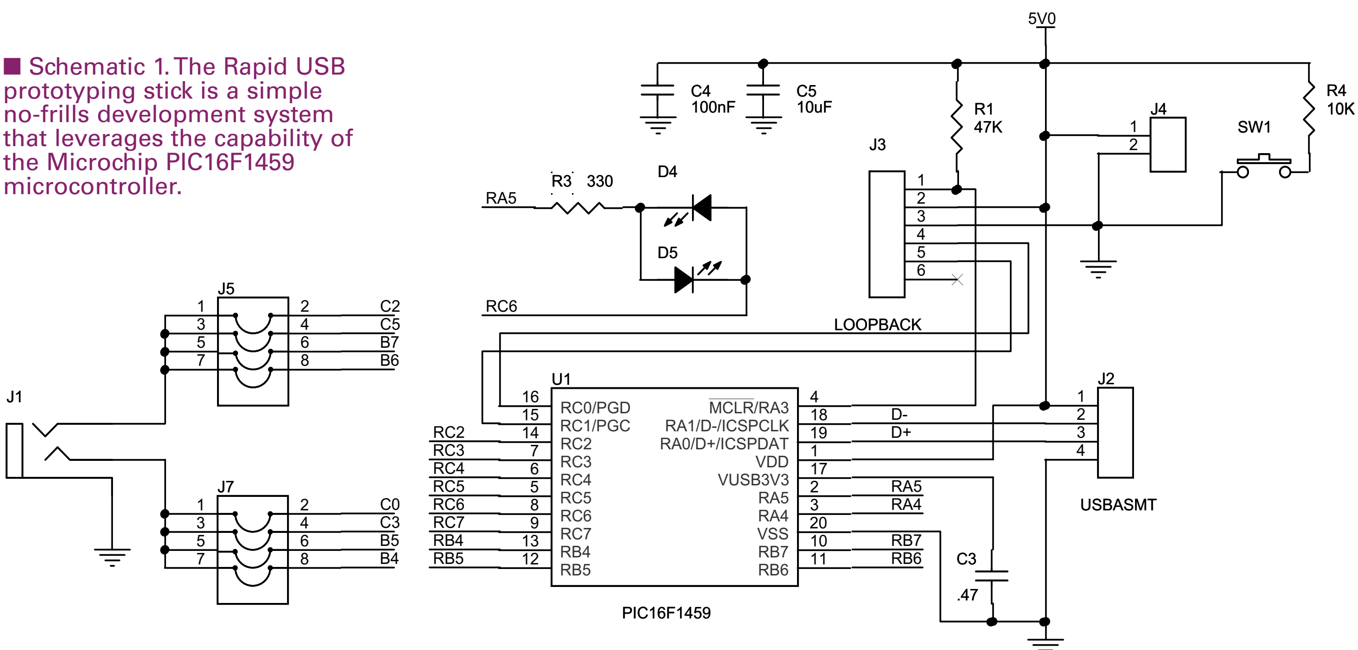
When I see "rapid" and "prototype" used in the same sentence, it gets my attention. That's why the plain white box you see in Photo 1 is holding court. The treasure that lies inside of this unassuming cardboard enclosure is yet to be discovered. So, let's break the seal on this mystery box and take a gander at the contents.

The CCS Rapid USB Prototyping Stick

Upon opening the box, I was greeted by the trio of objects shown in **Photo 2**. The contents of the box include a CD-ROM, a pluggable three-conductor cable, and a USB dongle sporting an integral pushbutton. In that I'm just a singer in a rock and roll band here, I followed the startup instructions that were conveyed on a "Read Me First" flyer that was found among the contents of the box. I proceeded to load the CD-ROM and kicked off the Rapid USB setup program.

I noticed that the Rapid USB setup operation wrote to my PICC directory, which is where my CCS C compiler is based. So, I navigated to the PICC directory. In search of anything related to Rapid USB within the PICC directory, I managed to find a PDF document and a file named *RapidUSB.h*. The PDF document (*Rapid USB Prototyping Stick Exercise Book*) contains an introduction to the Rapid USB development platform and numerous code examples.

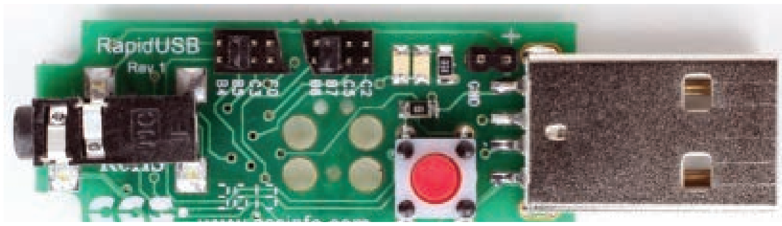
I examined the code examples to get a feel of what the Rapid USB platform could do. Armed with the knowledge of the firmware capabilities of the prototyping stick, I hit my fast forward button with the intention of locating the stick's schematic diagram. As you can see in **Schematic 1**, the Rapid USB prototyping stick is based on the Microchip PIC16F1459. The PIC16F1459 is an enhanced mid-range device. In other words, the PIC16F1459 can perform many of the functions that were previously reserved for more



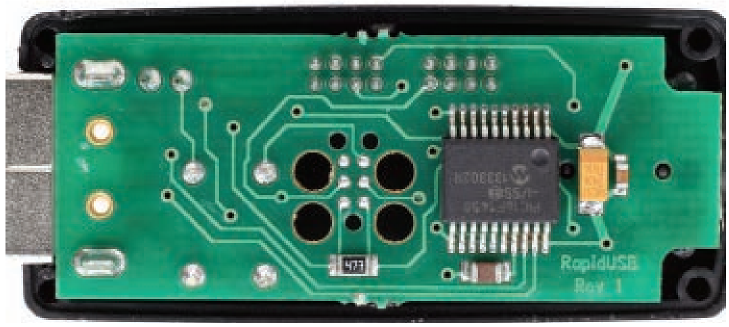
■ Schematic 1. The Rapid USB prototyping stick is a simple no-frills development system that leverages the capability of the Microchip PIC16F1459 microcontroller.

Post comments on this article and find any associated files and/or downloads at **www.nutsvolts.com/index.php?/magazine/article/november2014_DesignCycle**.

■ Photo 2. With the power of the CCS PIC C compiler behind them, this set of components is a formidable development platform and learning tool.



■ Photo 3. This is a top-side aerial view of the raw Rapid USB prototyping stick. The I/O selection headers are top-center. One I/O line on each four-position header is routed to the prototyping cable connector on the far left.



■ Photo 4. The PIC16F1459 is supported by only four external components. All of the USB and CPU clocking is handled internally by the microcontroller.

capable PIC18F devices. Standard RS-232 serial ports have been eliminated and replaced by USB portals on most — if not all — of the PCs on the market today. In terms of USB portals, microcontrollers large and small are following the PC lead. However, RS-232 capability remains a mainstay in the microcontroller world as many embedded device applications rely on RS-232 connectivity. Microcontrollers like the PIC16F1459 provide the best of both worlds employing on-chip USB functionality coupled with a native UART portal. Don't let the simplicity of this prototyping stick schematic fool you. The PIC16F1459 is capable of revealing alternate functions on selected I/O pins. For instance, I/O pin C1 can function as a comparator input and an external interrupt pin when it is not busy supporting the ICSP programming function. Depending on the pin, the PIC16F1459 I/O subsystem supports DAC, analog, SPI, I^2C, timer input, interrupt on change, UART, and PWM alternate functions.

I've popped the covers on our Rapid USB prototyping stick in **Photo 3**. As you can see, the component count is minimal. The LEDs in **Photo 3** are connected in parallel in such a way as to require only a single current-limiting resistor. The logic level applied to the LED pins determines if the red or green LED illuminates. The prototyping stick comes preloaded with a bootloader that is supported by the CCS C compiler IDE. So, don't be tempted to plug into the Rapid USB stick's Tag-Connect programming interface, which is located at dead center of its circuit board. The flip side of the prototyping stick is shown in **Photo 4**. In this view, you can clearly see the Tag-Connect's ICSP contacts. The six pads directly above the 47K resistor interface with the Tag-Connect cable's "bed of nails." The 47K resistor and the trio of capacitors are all that's needed to support the PIC16F1459.

The shorting blocks jumper-select one of the PIC16F1459 I/O lines on each of the four-position male headers. The pair of selected I/O lines is routed to the prototyping cable connector mounted to the far left of **Photo 3**. The prototyping cable shown in **Photo 2** is loaded with a red, white, and green conductor. The green conductor is grounded. The header pin/conductor logic is laid out in **Figure 1**. The pushbutton is electrically connected to the PIC16F1459's C1 pin. The advantage to this connection is that the pushbutton can be used to trigger the external interrupt pin. Naturally, the pushbutton position (open or closed) can also be sensed by reading the logic level presented to pin C1.

Header Position	White Conductor	Red Conductor
1	PIN_CO	PIN_C2
2	PIN_C3	PIN_C5
3	PIN_B5	PIN_B7
4	PIN_B4	PIN_B6

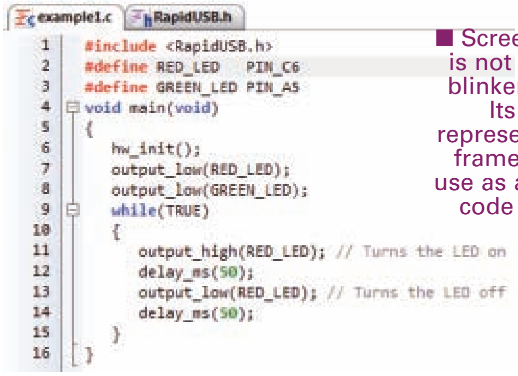
■ **Figure 1.** The header shunts in conjunction with the prototyping cable connector allow two of the PIC16F1459's I/O pins to be routed to an external destination.

Let's Blink an LED

It may seem immaterial, but learning to simply blink an LED provides important information about how to structure our applications. The code captured in **Screenshot 1** blinks the prototyping stick's red LED. If you are familiar with the CCS C compiler, you know that the Project Wizard will build an *include* file (*xxxx.h*) that is referenced at the top of your C source code. In the case

CCS
Rapid USB Prototyping Stick
CCS C Compiler
www.ccsinfo.com

Digi International
XBee
www.digi.com



```
#include <RapidUSB.h>
#define RED_LED    PIN_C6
#define GREEN_LED PIN_A5
void main(void)
{
   hw_init();
   output_low(RED_LED);
   output_low(GREEN_LED);
   while(TRUE)
   {
      output_high(RED_LED); // Turns the LED on
      delay_ms(50);
      output_low(RED_LED); // Turns the LED off
      delay_ms(50);
   }
}
```

■ Screenshot 1. This is not just a simple blinker application. Its components represent a working framework we can use as a template to code our projects.

of the Rapid USB prototyping stick, that *include* file exists in the form of *RapidUSB.h,* which is loaded into our CCS compiler directory by the prototyping stick setup program. You do not have to run the Project Wizard to construct Rapid USB projects. Everything is taken care of within *RapidUSB.h*. Just include the *RapidUSB.h* file and start writing your application.

The *hw_init* function is located within *RapidUSB.h* and should always be called first. An examination of the *hw_init* source code indicates that *hw_init* is actually a macro that only affects the comparator setup. With that, we could also write the LED blinker code in this manner:

```
#include <RapidUSB.h>
#define RED_LED       PIN_C6
#define GREEN_LED     PIN_A5

void init(void)
{
   hw_init();
   output_low(RED_LED);         //Both LEDs off
   output_low(GREEN_LED);
}

void main(void)
{
   init();
    while(TRUE)
   {
      output_high(RED_LED); // Turns the LED on
      delay_ms(50);
      output_low(RED_LED); // Turns the LED off
      delay_ms(50);
   }
}
```

Our revised code simply herds all of the initialization routines including *hw_init* into the *init* function. Compiling and loading our LED blinker program is as easy as clicking Build & Run. The bootloader process utilizes the services of the C compiler IDE, coupled with the built-in terminal emulator program (serial input/output monitor).

RS-232 and the Rapid USB Prototyping Stick

A quick look at the PIC16F1459 datasheet tells us that the TX and RX signals are found on pins RB7 and RB5, respectively. To direct the UART signals to the prototype cable connector, we must place I/O selection jumpers at positions B7 and B5. This jumper arrangement can be visualized by referencing **Photo 3**. The TX and RX signals are also identified in the prototyping stick's *Exercise Book*. Transmit signals will emanate from the red conductor of the prototyping cable, while receive signals enter on the white conductor. With the inclusion of the green conductor (ground), our prototyping cable represents a simple three-wire RS-232 TTL interface.

Since the prototyping stick's USB engine and UART are separate entities, we can write some code to form a communications link between them. The USB-to-UART link is great for processing the data stream. However, we routed the UART signals to the prototyping connector to expose the three-wire RS-232 port to external devices.

We cannot attach the Rapid USB stick's three-wire serial port directly to a regulation RS-232 port. True RS-232 signal levels fall in the ±15 VDC range. The prototyping stick's UART levels lie in the TTL voltage domain (0 VDC to +5.0 VDC). Attaching the stick to a regulation RS-232 port would release the magic smoke that is held within the PIC16F1459.

To verify the operation of the prototyping stick's UART, the most logical plan would be to use a laptop and a terminal emulator program to capture its transmission. Since my laptop is only equipped with USB portals, we must convert the prototyping stick's UART output stream to a USB-compatible data stream.

The easiest way to perform the conversion is to use an off-the-shelf USB-to-UART converter. I happen to have such a device. However, my USB-to-UART converter is designed to operate using 3.3 volt logic levels. The Rapid USB prototyping stick is a 5.0 volt module. Thus, the stick uses 5.0 volt logic levels. I've got an idea about the hardware. So, let's go ahead and write the RS-232 verification code:

```
#include <RapidUSB.h>
#define RED_LED       PIN_C6
#define GREEN_LED     PIN_A5

#use RS232(baud=9600, xmit=PIN_B7, rcv=PIN_B5,
stream=RADIO)

void init(void)
{
   hw_init();
   output_low(RED_LED);         //Both LEDs off
   output_low(GREEN_LED);
}

void main(void)
{
   init();
   while(TRUE)
   {
      fprintf(RADIO,"RapidUSB Toggle RED
      LED\r\n");
      output_toggle(RED_LED);
      delay_ms(500);
      fprintf(RADIO,"RapidUSB Toggle GREEN
      LED\r\n");
```

```
        output_toggle(GREEN_LED);
        delay_ms(500);
    }
}
```

The "wireless" USB-to-UART hardware configuration I had in mind is under the lens in **Photo 5**. I declared the prototyping stick's temporary independence from the laptop USB portal by soldering a four-pin 0.1 inch pitch male header to a USB 2.0 Type A female connector. As you can see in **Photo 5**, the female USB connector's outer pins carry the +5.0 VDC power rail which is provided by the host USB portal. Our RS-232 verification code is self-explanatory. We are simply sending a message and toggling the respective LED every 500 mS. All of the UART setup is performed in the *RapidUSB.h* code and the *#use RS232* pre-processor statement. Note that in structure, our enhanced RS-232 code does not differ greatly from our LED blinker code. Before the stick assumes its duty in **Photo 5**, the RS-232/LED blinker code is compiled and loaded into the stick in the normal fashion via the IDE and host laptop's USB portal.

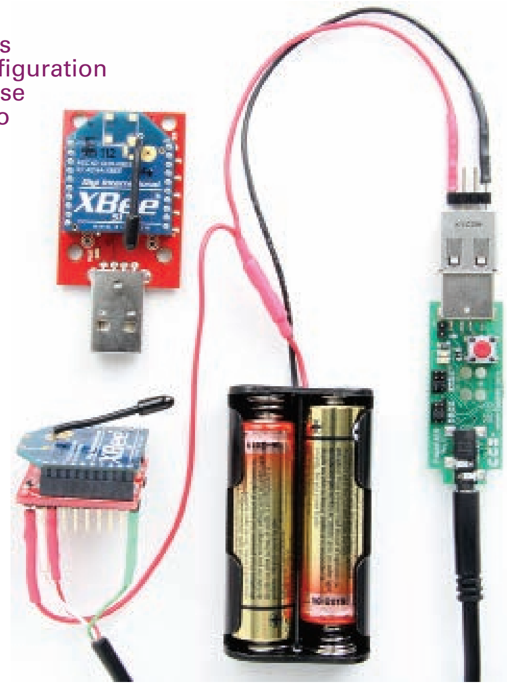
A pair of XBee data radios stands in place of a hard-wired USB-to-UART module and its associated USB cable. The XBee radio interface operates with 3.3 volt logic levels. As you can see in **Photo 5**, the prototyping stick and its attached XBee radio are powered by a +3.0 volt battery supply via the modified female Type A USB connector. This is just fine since the PIC16F1459 can operate at supply voltages between 1.8 and 5.5 volts. The +3.0 volt battery power also suits the XBee radio module as its recommended minimum operating voltage is 3.0 volts. At this point, a common and safe logic level of 3.0 volts is being used by the stick's UART and the attached XBee radio module.

The XBee radio mounted on the USB carrier is under control of a host USB portal on my laptop. Both the XBee radio that is attached to the RS-232 port of the prototyping stick and the XBee radio on the USB carrier board are programmed with out-of-the-box default settings. The default baud rate of the XBee radios is reflected in our verification code's *#use RS232* pre-processor statement.

The photographic flash units overpower the stick's LEDs which are blinking away in **Photo 5**. The stick's UART output data stream captured in **Screenshot 2** is presented by the CCS C compiler serial input/output monitor application, which is running in support of the XBee module that is plugged into my laptop's USB portal.

Sorry for the Interruption

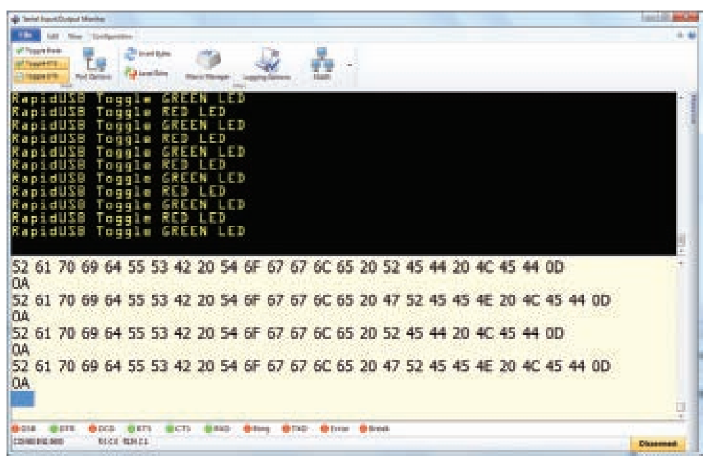
I'm dying to write some code aimed at that pushbutton! We will use the same wireless hardware configuration as seen in **Photo 5**. By the time we're done, the prototyping stick that we used to blink LEDs and drive XBee radios will be coaxed into a remote control mode. Let's begin by giving the stick's pushbutton a human name:



■ Photo 5. This hardware configuration allows us to use the XBee radio link as a USB-to-UART converter.

```
#include <RapidUSB.h>
#define RED_LED        PIN_C6
#define GREEN_LED      PIN_A5
#define PBTN           PIN_C1
```

The *PBTN* moniker will be used when the state of the pushbutton needs to be checked. The primary function of the pushbutton is to trigger an external interrupt which kicks off secondary processes. We will leave our prototyping stick UART setup code intact. However, we will need some additional variables to signal the interrupt trigger, hold our messages, and drive a mini state machine:



■ Screenshot 2. Powering the Rapid USB prototyping stick with a 3.0 volt battery pack allows us to interface the stick's UART to an XBee data radio module.

```
#use RS232(baud=9600, xmit=PIN_B7, rcv=PIN_B5,
stream=RADIO)

typedef struct{
   unsigned int8 fremote:1;
 }FFLAGS;
FFLAGS flags;

unsigned int8 msgON[3];
unsigned int8 msgOFF[4];
unsigned int8 aryIndex;
unsigned int8 nextState;
```

The *fremote* flag will be set within the pushbutton interrupt handler when the pushbutton is depressed. Normally, a remote control device will send a sequence of binary characters that are decoded at the receiving end. In this case, I'm going to use human readable messages which will be stored in the arrays *msgON* and *msgOFF*. Rather than use the mundane *i* or *x* as an array index pointer, we will use the *aryIndex* variable as our array pointer. Our remote control application is very simple and consists of two states (ON and OFF). The *nextState* variable will determine which state is to be entered with the next pushbutton depression.

Our pushbutton interrupt handler is to the point:

```
#INT_EXT
void  EXT_isr(void)
{
    flags.fremote = 1;
    clear_interrupt(INT_EXT);
}
```

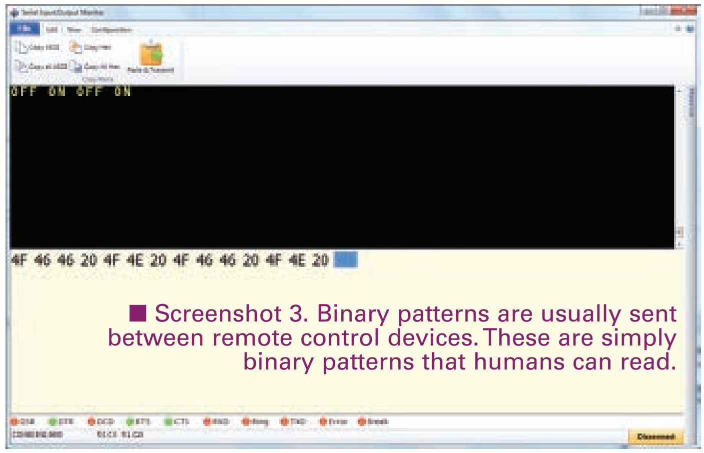
When the pushbutton is depressed, the *fremote* flag is set and the external interrupt flag is cleared. It is always best to do as little as possible within the interrupt handler. We will handle the pushbutton physics following the execution of the interrupt handler. Let's load our message arrays in the initialization function:

```
void init(void)
{
   hw_init();
   output_low(RED_LED);        //Both LEDs off
   output_low(GREEN_LED);

   msgON[0] = 'O';
   msgON[1] = 'N';
   msgON[2] = 0x00;

   msgOFF[0] = 'O';
   msgOFF[1] = 'F';
   msgOFF[2] = 'F';
   msgOFF[3] = 0x00;

   nextState = 0;

   flags.fremote = 0;
```

```
    enable_interrupts(INT_EXT_H2L);
}
```

The first state will be OFF, which is signaled by a zero in the *nextState* value. The final tasks we will perform within the *init* function include clearing the *fremote* flag and enabling the external interrupt. A quick look at the stick schematic reveals that the pushbutton I/O signal is pulled up with a 10K resistor. The *H2L* in the interrupt enable statement instructs the interrupt engine to fire when the pushbutton logic level transitions from logically high to logically low. The main loop of our prototyping stick remote control application consists of a C language switch mechanism that forms a mini state machine:

```
void main(void)
{
    init();
    while(TRUE)
    {
        switch(nextState)
        {
              case 0:
               if(flags.fremote == 1)
               {
                 disable_interrupts(INT_EXT);
                 delay_ms(100);
                 while(input(PBTN));
                 while(!input(PBTN));
```



■ Screenshot 3. Binary patterns are usually sent between remote control devices. These are simply binary patterns that humans can read.

```
for(aryIndex = 0;aryIndex <
3;aryIndex++)
 {
     fprintf(RADIO,"%c",msgOFF
     [aryIndex]);
 }
 fprintf(RADIO," ");
 output_high(RED_LED);
 delay_ms(500);
 output_low(RED_LED);
 nextState = 1;
 flags.fremote = 0;
 while(!input(PBTN));
```

```
            enable_interrupts(INT_EXT_H2L);
        }
        break;
```

The mini state machine runs continuously within the *while(TRUE)* loop. If the *nextState* value is equal to zero, the *case0* code executes. The *fremote* flag is checked to make sure that the pushbutton has indeed fired the external interrupt. To keep things from getting out of hand, we disable the external interrupt until the previously fired external interrupt is serviced.

The pushbutton is a mechanical device that bounces all over the place when depressed and released. Upon entering the *case0* code, we handle the pushbutton bounce by delaying 100 mS and waiting for the pushbutton to be released. If the bounce condition happens to satisfy the *while(input(PBTN))* and *while(!input(PBTN))* traps, the interrupt handler will execute normally and make another attempt to filter the bounce condition at the end of the interrupt handler code. The state machine does not move until the pushbutton is released.

The code within the *case0* state should be very familiar as it is based on our very first LED blinker application. The *case1* code is identical with the exception of the message contents:

```
    case 1:
        if(flags.fremote == 1)
        {
            disable_interrupts(INT_EXT);
            delay_ms(100);
            while(input(PBTN));
            while(!input(PBTN));
            for(aryIndex = 0;aryIndex <
            2;aryIndex++)
            {
                fprintf(RADIO,"%c",msgON
                [aryIndex]);
            }
            fprintf(RADIO," ");
            output_high(GREEN_LED);
            delay_ms(500);
            output_low(GREEN_LED);
            nextState = 0;
            flags.fremote = 0;
            while(!input(PBTN));
            enable_interrupts(INT_EXT_H2L);
        }
        break;
      }
   }
}
```

I have captured the results of depressing and releasing the pushbutton in **Screenshot 3**.

Designing Outside of the Box

The Rapid USB prototyping stick provides everything you need to quickly assemble some rather powerful applications with little effort and minimal extraneous hardware. **NV**

The Arduino Classroom Arduino 101 — Chapter 10: Sensing Light and Temperature

We've learned that computers can sense their environment and — based on what is sensed — they can make decisions and control things. Simply stated, computers sense, decide, and act. We learned in Chapter 4 how to sense a button press and then based on that information, chose to act by turning an LED on or off. Your PC senses keyboard button presses using techniques similar to what we've learned, and then makes decisions and acts. For instance, if you are using a word processing program, the PC senses that you press the 'a' button; it then "decides" what it needs to do, and "acts" by displaying an 'a' on the screen. While that's a very simple example, the basic principles of sensing, deciding, and acting represent the same sequence used by more complex computing and electronics systems. When one of Google's autonomous cars drives itself through San Francisco — a very complex task — it is simply sensing, deciding, and acting. This month, we will begin expanding what we can sense with the Arduino by learning to sense light and temperature.

Humidity

One of the early discoveries in electricity was that some materials have a resistance that varies with the amount of light that falls on them. **Cadmium Sulfide (CdS)** is one such material. This **light dependent resistance** was used to measure light levels for photography. Today, this and similar material are used in a variety of applications, including turning on lights at dusk and then turning them off again at dawn. We also see them in sensors in entertainment devices that detect infrared light, and are used to receive commands from handheld remote controls. Digital cameras have millions of light sensors to detect the red, green, and blue light falling on each sensor picture element.

■ FIGURE 1: *CdS light dependent resistor photo.*

■ FIGURE 2: *CDS sensor breadboard and schematic symbols.*

Our CdS light sensor (**Figure 1**) has a resistance that is inversely proportional to the incident light. The higher the light, the lower the resistance. We can use this by applying what we learned in Chapters 6 and 7 about utilizing resistors as voltage dividers.

In Lab 2 of Chapter 7, we discovered that if we placed a potentiometer (a variable resistor) in series with a constant resistor, we could measure the voltage change due to the potentiometer position. When we vary the angle of the potentiometer (thus changing its resistance), then the current flowing into the constant resistor depends on that current variation. This variation in current changes the voltage across the constant resistor which we then measure using an ADC (analog-to-digital converter) and the *analogInput()* command. The light sensor works much like the potentiometer, so it can be placed in series with a constant resistor.

We can measure the voltage across that resistor as an

Post comments on this article and find any associated files and/or downloads at **www.nutsvolts.com/index.php?/magazine/article/november2014_Smiley**.

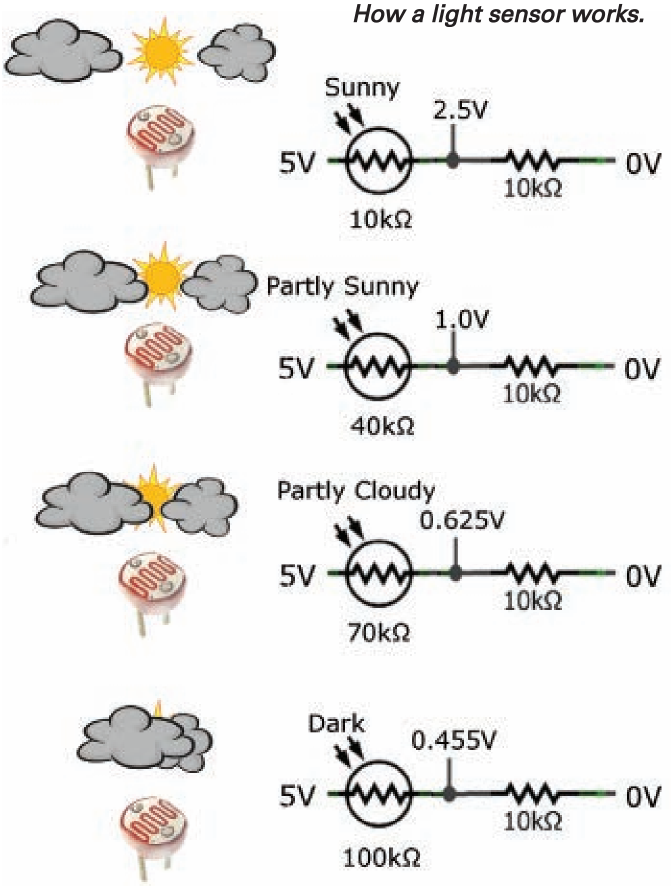


FIGURE 3: *How a light sensor works.*

indicator of the changing resistance of the light sensor.

As we see in **Figure 3**, the voltage will vary because the resistance is varying in the sensor, thus acting as a sort of valve controlling the current through the 10K Ω resistor. As we did in Chapter 8, we measure the voltage across the 10K Ω resistor and find that this value is proportional to the light falling on the sensor.

To further illustrate this, I built the circuit shown in Lab 1 and placed it under a very bright light. I then used my hand to shade the sensor, moving my hand up and down to vary the light falling on it. I got the following voltages:

Covered:	0.415 volts
2 inches:	2.222 volts
4 inches:	2.627 volts
6 inches:	2.964 volts
8 inches:	3.311 volts
10 inches:	3.374 volts
Uncovered:	3.877 volts

You can clearly see that we are getting measurable voltage variations from the varying light falling on the sensor. We can use those voltages to control a device, much like in Chapter 8 where we used the varying voltages from a potentiometer to control the brightness of an LED or the angle of a servomotor. If we had a very bright light (such as the sun at noon), I expect we would get significantly different results from those shown here, but there would still be a proportional relationship.

Let's reinforce our learning about Ohm's Law to calculate the resistance across the light sensor at each of the readings:

Current = Voltage/Resistance (I = V/R)

Covered:	Current = 0.415/10000 = 0.0000415 amps
2 inches:	Current = 2.222/10000 = 0.0002222 amps
4 inches:	Current = 2.627/10000 = 0.0002627 amps
6 inches:	Current = 2.964/10000 = 0.0002964 amps
8 inches:	Current = 3.311/10000 = 0.0003311 amps
10 inches:	Current = 3.374/10000 = 0.0003374 amps
Uncovered:	Current = 3.877/10000 = 0.0003877 amps

This tells us what portion of the five volts is dropped across the 10K Ω resistor. The rest of that five volts is dropped across the light sensor, so we calculate the voltage across it as:

Covered:	Sensor voltage = 5 - 0.415 = 4.585
2 inches:	Sensor voltage = 5 - 2.222 = 2.778
4 inches:	Sensor voltage = 5 - 2.627 = 2.373
6 inches:	Sensor voltage = 5 - 2.964 = 2.036
8 inches:	Sensor voltage = 5 - 3.311 = 1.689
10 inches:	Sensor voltage = 5 - 3.374 = 1.626
Uncovered:	Sensor voltage = 5 - 3.877 = 1.123

Then, using Ohm's Law, we get the light sensor resistance:

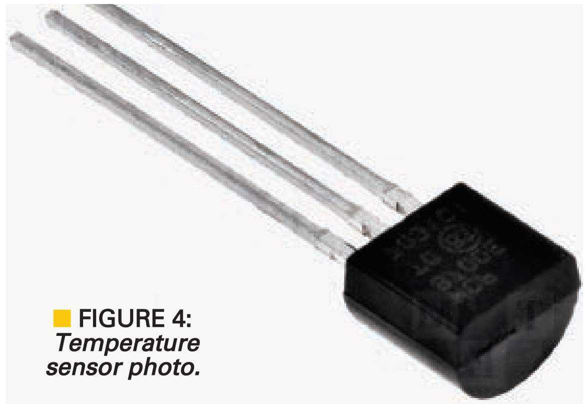
Sensor Resistance = Voltage/Current (R = V/I)

Covered:	Sensor resistance = 4.585/0.0000415 = 110,482 Ω
2 inches:	Sensor resistance = 2.778/0.0002222 = 12,513 Ω
4 inches:	Sensor resistance = 2.373/0.0002627 = 9,033 Ω
6 inches:	Sensor resistance = 2.036/0.0002964 = 6,869 Ω
8 inches:	Sensor resistance = 1.689/0.0003311 = 5,101 Ω
10 inches:	Sensor resistance = 1.626/0.0003374 = 4,819 Ω
Uncovered:	Sensor resistance = 1.123/0.0003877 = 2,897 Ω

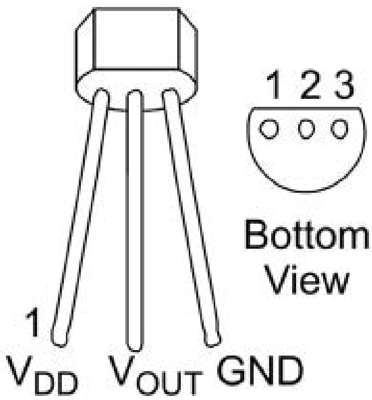
So, what this is telling us is that when the light sensor is exposed to very little light, the resistance is over 110,000 Ω, and that just a little extra light (hand at two inches) drops that resistance to a bit over 12,500 Ω. The

resistance progressively drops as more light is added (the hand is raised), bringing it down to about 2900 Ω when the light sensor is uncovered. This doesn't really tell us anything about the absolute value of that light as measured in some accepted dimension such as lux, but it does tell us that we are able to get a relative change in voltage as a function of changes in light level due to moving our hand. We will use these facts in our labs.

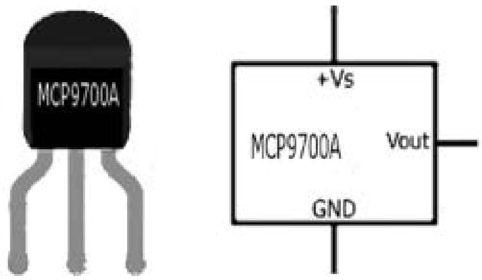
Sensing Temperature With a Thermistor IC



■ **FIGURE 4:** ***Temperature sensor photo.***



■ **FIGURE 5:** ***Temperature sensor datasheet drawing.***



■ **FIGURE 6:** ***Temperature sensor breadboard and schematic symbols.***

A thermistor is a resistor whose resistance varies with temperature. This is similar to the light sensor that varies resistance with light levels. The Arduino 101 Projects Kit has a MCP9700A thermistor IC (Integrated Circuit) that contains signal-conditioning circuitry that outputs a voltage proportional to the change in temperature. This circuitry simplifies our task since we can use the Arduino ADC to measure the voltage and directly translate the reading to a temperature.

The IC uses a T0-92 package as shown in **Figure 4**. **Figure 5** shows a drawing from the datasheet that illustrates the pin locations. **Figure 6** has the breadboard and schematic symbols we will use. This device measures in Celsius, and its accuracy is ±2°C from -40°C to 125°C. For those of us in America, we translate Celsius to Fahrenheit in our lab software using the following formula: °F = °C x 9/5 + 32.

The MCP9700A outputs 10.0 mV/°C (10.0 millivolts [.01 volts] per degree Celsius) scaled for 500 mV output at 0°C. Thus, if we measure 500 mV (0.5V) on the ADC, we know the sensor is detecting 0°C. We can use this base point (500 mV for 0°C) to calculate the voltage for the maximum accurate temperature of 125°C, and the minimum accurate temperature of -40°C.

Stop for a moment and think about how you would do this. First, look at the maximum accurate temperature: 125°C. We know that we have 500 mV for 0°C and that the sensor outputs 10 mV per °C. This means we have 10 mV per °C for 125°C, so we will see an additional 10*125 = 1250 mV above the mV level for 0°C.

We add the mV at 0°C/500 mV to the 1,250 mV for 125°C, and calculate that we would read 1,750 mV as the voltage for our maximum accurate temperature reading. If this isn't clear, then please write it out and do the calculations by hand so you understand how to convert from volts to temperature.

Let's repeat this process to determine the voltage for the minimum accurate temperature, which is -40°C. See if you can do this before reading further. We have a total of -40°C, which would equal a -400 mV change. Since we know that we would read 500 mV for 0°C, then we can subtract the 400 mV to get a -40°C reading of 500 - 400 = 100 mV. We now know that if we read 100 mV, we have -40°C; if we read 500 mV, we have 0°C; and if we read 1750 mV, we have 125°C. Got it?

We saw in Chapters 6 and 7 that our ADC looks at voltages from zero to five volts, and divides that into 1024 steps with 0 being zero volts and 1023 being five volts. We further saw that each ADC step represents 5/1024 = 0.00488 volts (4.88 mV). We know that the sensor outputs 10 mV per °C, so how many steps in the ADC output would indicate 10 mV?

We see that the 10 mV divided by 4.88 mV gives us 2.04918 steps, but the ADC steps are integers so we are going to have to do some math here. Two steps of the ADC (2 * 4.88 mV) equal 9.77 mV, so two steps are pretty close to a single °C change in temperature (10 mV). Notice that the error between 10 and 9.77 is just under 1%, and since the error for the sensor itself is 2% we might be safe ignoring it and saying that two steps in the ADC output is equal to exactly 1°C.

Doing that would allow us to use integers rather than floating point data (a data type that has integer and fractional parts divided by a decimal point), which uses a

lot more of the computer's resources to do calculations that do integers. Using the integer simplification would probably be good enough for most applications and would save us some serious computing. However, since we aren't doing a lot of extra work with this program, we will go ahead and write the code using floating point data types. (To rephrase: In a more constrained program that does a lot of work and takes up a lot of space, we probably wouldn't use floating point, but accept the error and say that two ADC steps equals 1°C.)

Or, saying this another way, we know that the temperature sensor will output voltages between 100 mV and 1,750 mV to indicate -40°C to +125°C. The ADC outputs values from 0 to 1023 in steps representing 4.88 mV per step. This tells us that the lowest valid ADC reading will be for 100 mV giving us: 100 mV/4.88 mV/step = 20.49 steps. Of course, these steps are integers so we will say that the lowest valid step value is 21, which equals 21 steps * 4.88 mV/step = 102.48 mV.

We know the temperature sensor outputs 100 mV for -40°C, so 102.48 mV is slightly above that. We know that the sensor outputs 10 mV per °C, so we can calculate 2.48 mV / 10 mV/ °C = .248°C, giving us a temperature of -40°C + .248 °C = -39.75°C. Of course, our real accuracy isn't quite that good, but for our purposes here, we'll say that our systems lowest accurate temperature is -39.75°C.

We can repeat this logic to calculate the highest accurate temperature reading: 1750 mV / 4.88 mV/step = 358.61 steps (359 steps since it is an integer). However, when we calculate 359 steps * 4.88 mV/step = 1751.92 mV, we see that the 359 takes us above the highest accurate temperature output (that is, 125°C at 1750 mV), so let's drop the maximum step to 358 and calculate 358 steps * 4.88 mV/step = 1747.04 mV, which is in range.

We now see that the range of accurate readings from our ADC will be 21 to 358 out of a possible range of 0 to 1023. For our purposes, we should consider any ADC readings below 21 or above 358 as invtalid, and ignore them when we do our software conversions of the ADC readings to voltage and then to °C.

Since this discussion is fairly complex, we will have a lab to use all of this in a test program to convert fake ADC readings to voltage and °C.

Lab 1: Sensing Relative Light Levels

This lab implements the light measurement discussed previously. This circuit works like the one back in **Figure 3**, except we will use our hand instead of clouds to vary the light falling on the sensor. We will then use the Arduino ADC to measure the changing voltage across the resistor.

Parts required:
1 Arduino
1 USB cable
1 Arduino proto shield and jumper wires
1 CdS light sensor
1 10,000 Ω resistor

Estimated time for this lab: 30 minutes

Check off when complete:
❑ Build the circuit shown in **Figures 7**, **8**, and **9**.
❑ Load the following program into the Arduino IDE

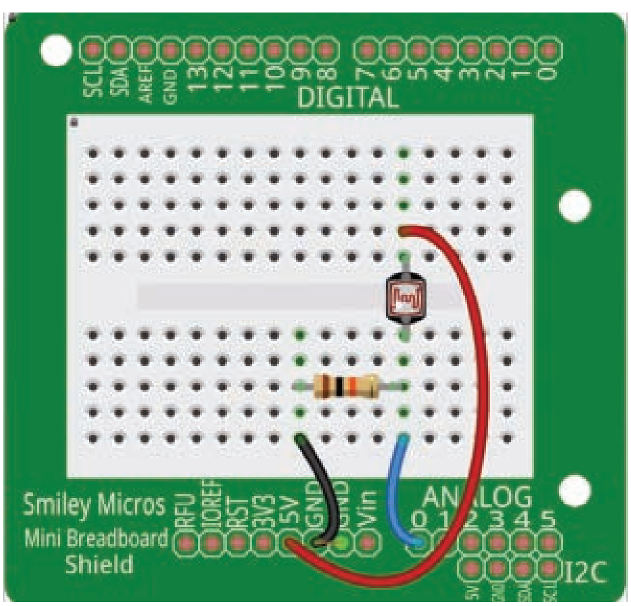


FIGURE 7: ***CdS sensor circuit breadboard.***

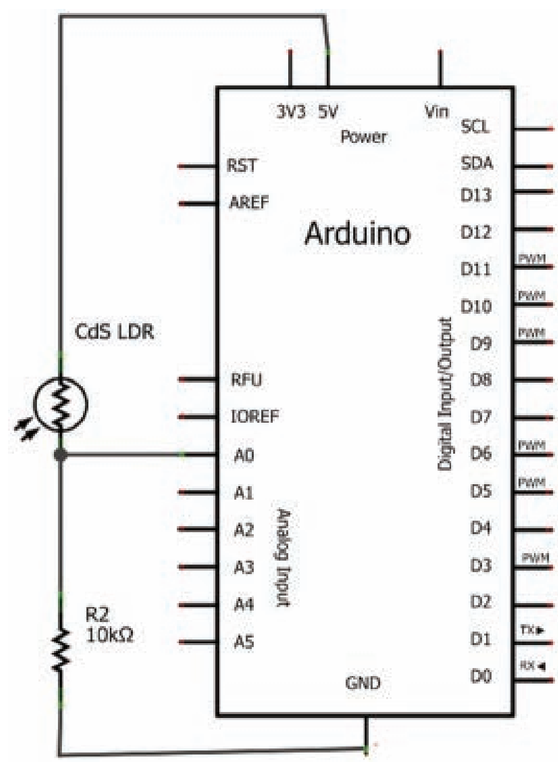


FIGURE 8: ***CdS sensor circuit schematic.***

(go to the article link to find files for all of the different code listings):

```
// A101_ch10_light_sensor_voltage 7/2/14
// Joe Pardue

int sensorPin = A0;   // analog input pin
int sensorValue = 0;  // store the analog input
                      // value

void setup() {
  Serial.begin(57600);
  Serial.println("Measure light sensor voltage
  rev 1.0");
}

void loop() {

  if(Serial.available())
  {
    char c = Serial.read();
    if(c == 'r')
    {
      // read the value from the sensor:
      sensorValue = analogRead(sensorPin);

      Serial.print("LDR voltage: ");
      Serial.print(sensorValue);
      Serial.print("  Voltage: ");

Serial.println(((5.0*(float)sensorValue)/1024.0),
3);
    }
  }
}
```

❑ Notice that this program is identical to the *A101_ch7_pot_voltage.ino* program from Chapter 7, except that some comments are changed. This powerfully illustrates the value of code reuse.
❑ Load the program onto your Arduino and then run the Arduino serial monitor.
❑ Repeat the earlier experiment where you cover the sensor with your hand and then move it roughly two inches up for each subsequent measurement.
❑ You should get results that appear somewhat like those in **Figure 10**.

Lab 2: Converting ADC Sensor Readings to Temperature

Converting Arduino ADC readings to voltage and then converting those voltages to temperature readings is somewhat complicated. To reinforce our earlier discussion, we will write a program to create a series of fake ADC readings and verify that the techniques we discussed work as described.

Parts required:
1 Arduino
1 USB cable

Estimated time for this lab: 30 minutes

Check off when complete:
❑ Load the following program into the Arduino IDE:

```
// A101_ch10_volts_to_temperature_test 8/14/14

#define VOLTS_PER_ADC_STEP 5.0/1024.0
#define VOLTS_PER_DEGREE 0.01
#define temp_BASE -40.0
#define LOW_ADC 20.49
#define HIGH_ADC 358.61
#define LOW_VOLTAGE 0.5

int tempSensorPin = A0;     // analog input pin
float tempSensorADC = 0.0; // variable to for
                           // the ADC reading
float tempertureSensorVoltage = 0.0;
   // variable for the converted voltage
float tempSensorVoltage = 0.0;
   // variable for the sensor voltage
float tempCelsius = 0.0;
   // variable for degrees Celsius
```

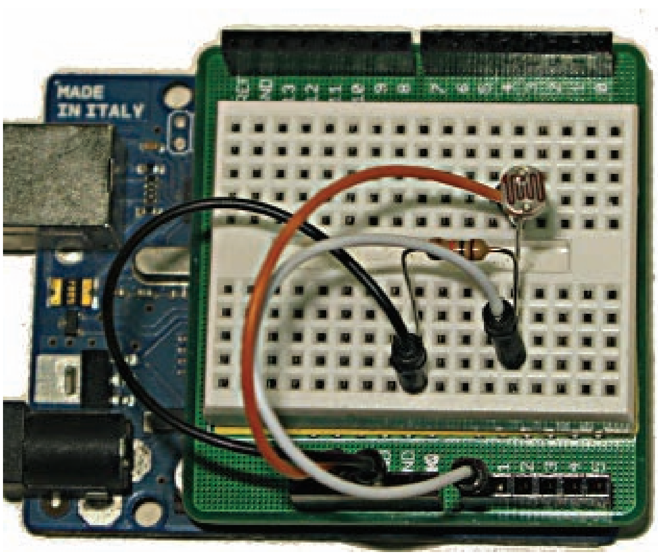


FIGURE 9: *CdS sensor circuit photo.*

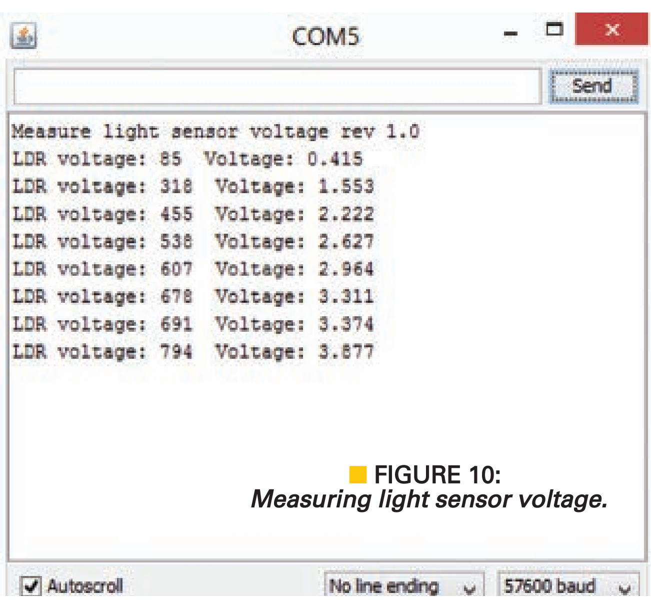


FIGURE 10: *Measuring light sensor voltage.*

```
void setup()
{
  Serial.begin(57600);
  Serial.println("Volts to temp test. rev 1.0");

 // Look from lowest ADC reading in +10 steps
 for(float i = LOW_ADC; i < HIGH_ADC; i += 10.0)
 {
  tempSensorADC = i;
  tempSensorVoltage = (tempSensorADC *
VOLTS_PER_ADC_STEP);
  tempCelsius = ((tempSensorVoltage-0.5) /
  VOLTS_PER_DEGREE);

  showIt();
 }

  // Look at highest acceptable ADC reading
  tempSensorADC = HIGH_ADC;
  tempSensorVoltage = (tempSensorADC * VOLTS_PER
  _ADC_STEP);
  tempCelsius = ((tempSensorVoltage - 0.5)/
  VOLTS_PER_DEGREE);
  showIt();
}

void loop()
{
  // do nothing
}temp

void showIt()
{
  Serial.print("tempSensorADC = ");
  Serial.println(tempSensorADC);
  Serial.print("tempSensorVoltage = ");
  Serial.println(tempSensorVoltage);
  Serial.print("Temperature in Celsius = ");
  Serial.println(tempCelsius);
  Serial.print("Temperature in Fahrenheit = ");
  // °F = °C  x  9/5 + 32
  Serial.println( ((tempCelsius * 9)/5) + 32);
}
```

❑ Compile and run the program, and verify that you get the output shown in **Figure 11**.

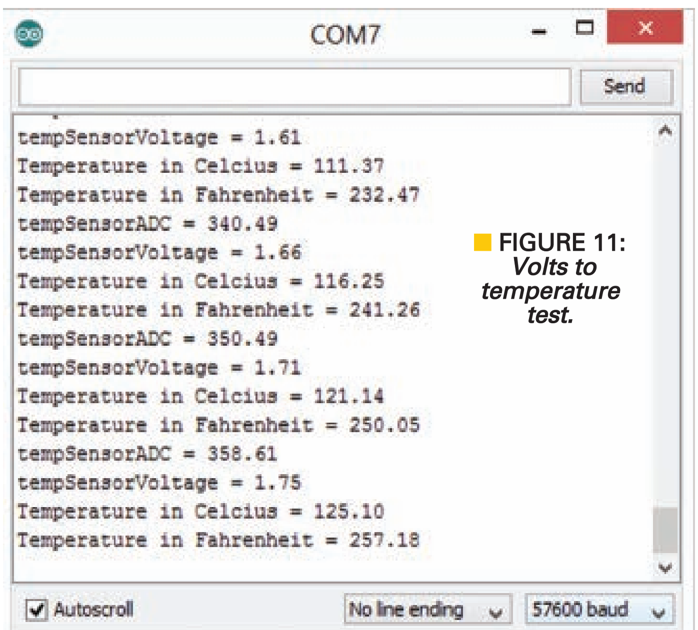


FIGURE 11: *Volts to temperature test.*

❑ Make sure you are clear on what is going on in this program. If necessary, review the earlier discussion and the source code.

Lab 3: Sensing Temperature

For this lab, we will reuse the circuit built for the previous labs and add the temperature sensor to it. We won't use the light sensor in this lab, but in the next chapter we will use both sensors together.

Parts required:

1 Arduino
1 USB cable
1 Arduino proto shield and jumper wires
1 CdS light sensor
1 10,000 Ω resistor
1 MCP9700A temperature sensor

Estimated time for this lab: 30 minutes

Check off when complete:

❑ Build the circuit shown in **Figures 12** and **13**.
❑ Load the following program into the Arduino IDE:

```
// A101_ch10_sensing_temperature 8/15/14

#define VOLTS_PER_ADC_STEP 5.0/1024.0
#define VOLTS_PER_DEGREE 0.01
#define temp_BASE -40.0
#define LOW_ADC 20.49
#define HIGH_ADC 358.61
#define LOW_VOLTAGE 0.5

int tempSensorPin = A1;   // analog input pin for
                          // temperature sensor
```

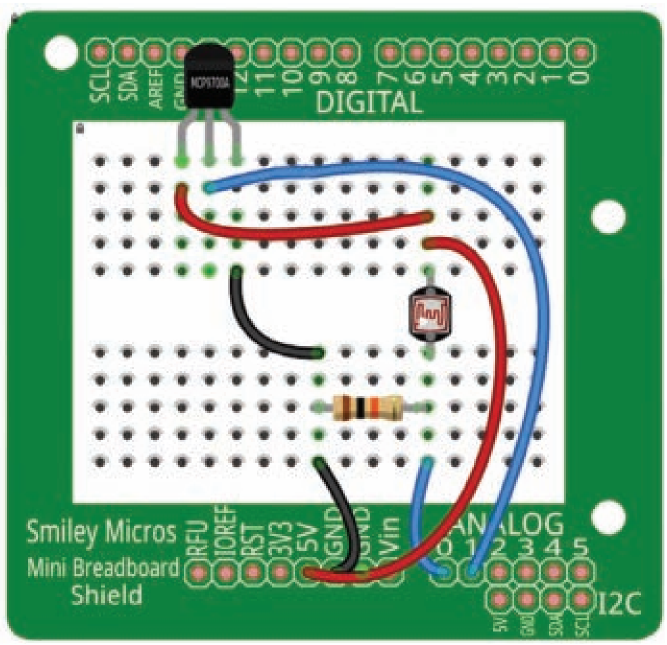
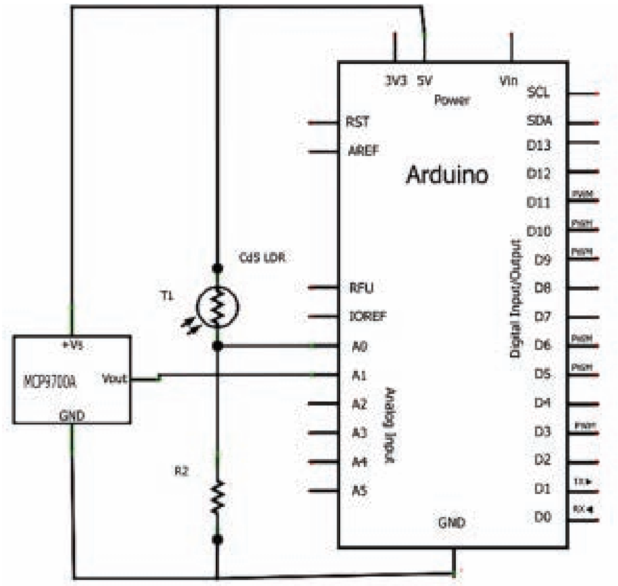
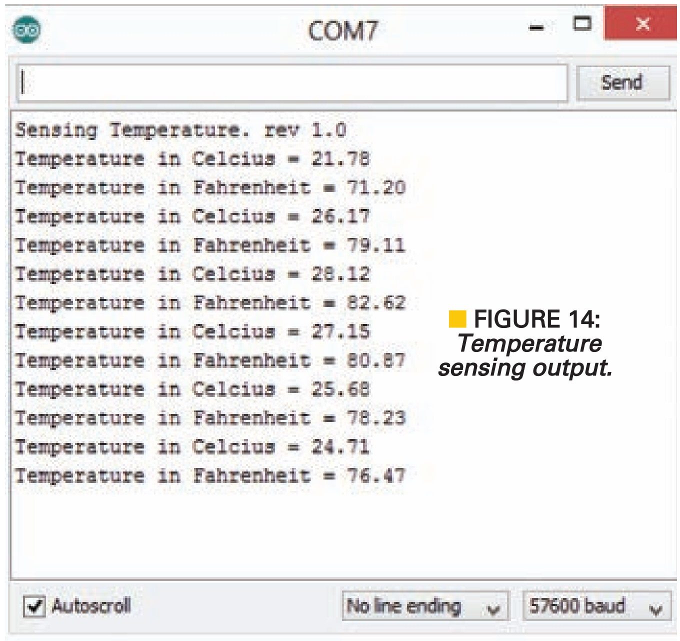


FIGURE 12: *Temperature sensor breadboard.*



■ FIGURE 13:
Temperature sensor schematic.



■ FIGURE 14:
Temperature sensing output.

```
float tempSensorADC = 0.0;
    // variable to for the ADC reading
float tempertureSensorVoltage = 0.0;
  // variable for the converted voltage
float tempSensorVoltage = 0.0;
  // variable for the sensor voltage
float tempCelsius = 0.0;
  // variable for degrees Celsius

void setup()
{
  Serial.begin(57600);
  Serial.println("Sensing Temperature. rev
  1.0");
}

void loop()
{
  tempSensorADC = analogRead(tempSensorPin);
  tempSensorVoltage = (tempSensorADC * VOLTS_
  PER_ADC_STEP);
  tempCelsius = ((tempSensorVoltage - 0.5)/
  VOLTS_PER_DEGREE);

  Serial.print("Temperature in Celsius = ");
  Serial.println(tempCelsius);
  Serial.print("Temperature in Fahrenheit = ");
  // °F = °C  x  9/5 + 32
  Serial.println( ((tempCelsius * 9)/5) + 32);

  delay(5000);
}
```

❏ Compile and run the program.
❏ After the first temperature reading, gently squeeze the sensor between your thumb and forefinger. Hold the sensor for several readings, then release it. Verify that you get a temperature rise and fall output similar to that shown in **Figure 14**.

Next month, we will combine a lot of what we have learned so far to create a data logger. **NV**

ELECTRONET

For the ElectroNet online, go to
www.nutsvolts.com
click *Electro-Net*

Video Monitoring Over the Internet

Affordable remote viewing with your smartphone.

Video has gotten so good and so cheap, everyone is using it. People are regularly posting their cell phone videos on YouTube and sending them as texts and emails. Others are taking action videos in HD using one of the small popular video cams like the GoPro. And who isn't streaming Netflix videos these days? There is one more growing video trend: *video monitoring*. It is now easy to do and very economical. Thanks to the Internet of Things (IoT) movement, you can do this yourself affordably.

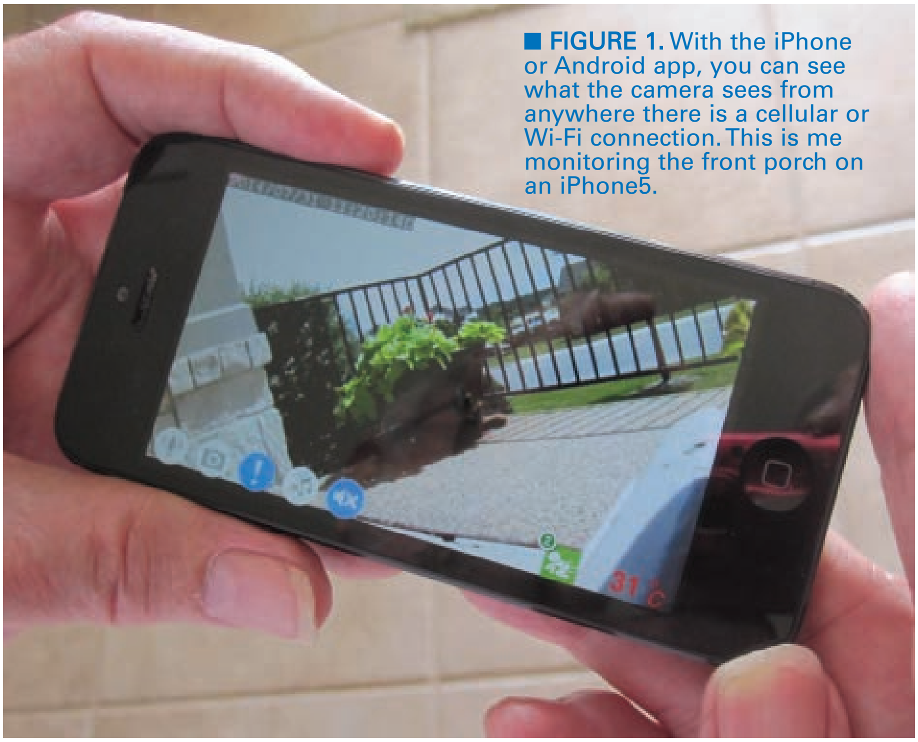
HOW IT WORKS

Video monitoring is also known as video surveillance. You've already experienced it yourself as you have passed one of the millions of closed circuit surveillance cameras mounted everywhere these days. Someone may be watching a monitor, but more likely the camera feed is being recorded for future viewing. Such cameras are usually connected by coax or twisted pair cable to the surveillance center, but more and more wireless links are being used. Many are now using the Internet to monitor things remotely. You can do that, too. What used to be expensive and tricky is now common place and affordable.

Remote home monitoring can be achieved with a low cost camera and a remote viewing device like a smartphone, tablet, or laptop. The camera links to your home Wi-Fi router and connects it to the Internet by way of your cable TV or DSL connection. Then, using an app, the smartphone miraculously connects to your home router via your cellular carrier and Internet provider, and voila'! You can see what your camera is seeing. **Figure 1** shows my iPhone looking at the camera feed outside my home.

HOW TO DO IT

My experience is with the D-Link DCS-825L baby camera (**Figure 2**). It is sold as a baby monitor, but you



■ FIGURE 1. With the iPhone or Android app, you can see what the camera sees from anywhere there is a cellular or Wi-Fi connection. This is me monitoring the front porch on an iPhone5.

Go to **www.nutsvolts.com/index.php?/magazine/article/november2014_OpenComm** to comment on this article.

■ **FIGURE 2.** The D-Link DCS-825L Internet camera is sold as a baby monitor, but you can use it for many other applications. It includes IR dark viewing, as well as two-way voice communications.

can use it for any monitoring application. Some of the features of this unit are:

- Day or night vision. Camera has infrared (IR) lights that illuminate a dark room and the camera has IR sensing. It can see up to 16 feet in total darkness.
- Motion detection. Alerts you if motion is detected in the viewing range.
- Digital zoom remotely and manual lens focus.
- Sound detection. Camera also contains a microphone that picks up any audio in the viewing area.
- Two-way audio. Unit has a speaker that lets you talk to someone in the viewing range via your smartphone or tablet.
- Temperature measurement. The antenna-like appendage on the camera in **Figure 2** is a temperature sensor that lets you remotely view the temperature in the camera environment.
- All camera features and settings can be controlled remotely on a cell phone.
- Mounting hardware is provided if you need it.

Most other Internet cameras have similar features, but a lot are bare-bones units with no frills. Check before you buy.

To set up the camera, first download the app. For this camera, there is both an Apple iPhone app and one for Android phones. Go to the Apple App Store or Google Play to get the free app. Open it and begin the installation process. Step-by-step directions are given to link the camera to your local Wi-Fi network. Once the setup is complete, click on the appropriate icon and begin monitoring. You will probably have to adjust the focus manually to get a clear view, but once it is set you are good to go. The video quality is excellent. The unit is powered with an AC wall transformer, but an optional battery power accessory is available if you need it.

Remote home monitoring can be achieved with a low cost camera and a remote viewing device like a smartphone, tablet, or laptop.

While no babies were involved with my camera, my wife and I have used it to check on the house while we were away for the weekend. We were able to discover what was mashing the flowers on the front porch during the night. The camera clearly picked up the stray cat doing the damage. Overall, the camera gives you some sense of security and control when you are not at home.

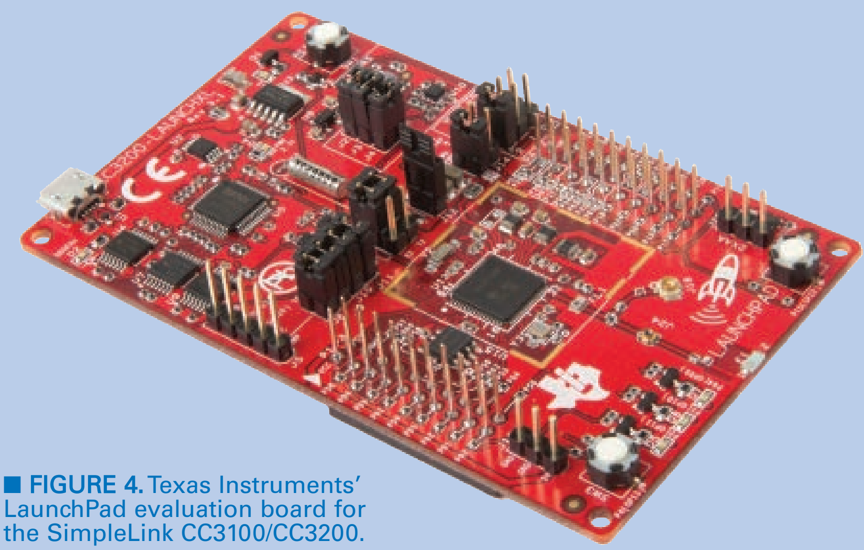
WHAT IS AVAILABLE?

The D-Link unit I used cost under $200, but it is a high-end unit. You can easily find lower cost units. Here are a few other Internet cameras I have discovered. One widely promoted camera is the Dropcam (**www.dropcam.com**). It comes in two versions: one with a 130 degree field of view and an 8x zoom for $199; and a 107 degree field of view with 4x zoom for $149. It too has both iPhone and Android apps. Samsung also has a comparable camera called the SmartCam. Both the Dropcam and Samsung units are 1080p HD. Another source is X10 – the company that makes a variety of home automation gadgets. They have some low cost video cams for around the $100 range. Check them out at **www.x10.com**. For lots of resources on video cameras and surveillance, go to **www.ipmv.com**

CONNECTING ANYTHING TO THE INTERNET

Connecting almost anything to the Internet is getting easier as practically all semiconductor and electronics companies are seeking

■ **FIGURE 3.** Texas Instruments' SimpleLink CC3100/CC3200 ICs are an Internet-on-a Chip, making IoT easy.



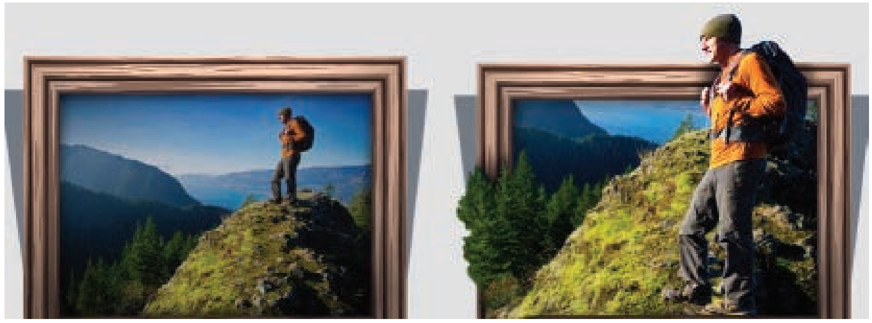
■ **FIGURE 4.** Texas Instruments' LaunchPad evaluation board for the SimpleLink CC3100/CC3200.

ways to participate in the IoT movement. A good example is RadioShack's forthcoming move to supply kits that help consumers do the IoT thing. They want to showcase and sell products that will turn things off and on remotely with simple equipment. Some of these products should be in a few of the stores now.

As for semiconductors, Texas Instruments recently introduced its new SimpleLink Wi-Fi CC3100 and CC3200 platform for IoT applications (refer to **Figure 3**). This new Internet-on-a-chip™ family allows you to easily add embedded Wi-Fi and Internet to a wide range of home, industrial, and consumer electronics. A key feature is very low power consumption for battery operated devices. The CC3100 provides the flexibility to use any microcontroller (MCU). The CC3200 has an integrated programmable ARM® Cortex®-M4 MCU that allows you to add your own code. These products offer the ability to simply and securely connect your devices to Wi-Fi using a phone or tablet app or a web browser with multiple provisioning options. TI offers the chips in QFN packages, as well as a wide range of evaluation boards (**Figure 4**), development kits, and support software. **NV**

NEW PRODUCTS

Continued from page 23



TRANSFER PERSONAL PHOTOS INTO 3D

Captivate family and friends with the experience of viewing traditional photos changed into 3D with a new process offered by Trummy.

By launching its high quality proprietary 2D-3D conversion service, Trummy makes it possible to transform digital photos into 3D experiences. Photos can be viewed on any device, anywhere, at anytime.

Trummy has combined a dedicated team of 3D visual artists who work with the proprietary Trummy technology that makes the process affordable.

Current 2D-3D photo conversion services are either based on automatic algorithms that generate poor quality images or are made by artists who costs are typically out of reach for most consumers.

To utilize the service, begin by registering at their website, upload photos, then select the ones to be converted.

Trummy's 3D visual artists will evaluate the potential of each photo in 3D to make sure the best possible 3D experience will be achieved.

After evaluation, the 3D visual artist starts to create the 3D photo. This manual conversion process will take a maximum of 48 hours.

As soon as the photo is converted, notification is sent via email.

At the moment, Trummy is offering this service for $3.50 per photo. The image can be downloaded in all popular 3D formats, making it accessible on any device.

For more information, contact:
Trummy
Web: **www.trummy.com**

 Continued from page 7

resistive divider to get a split supply, I stepped up to the plate and used a TI rail splitter part (part #TLE2426). This IC works exactly as advertised and is only the size of a TO-92 transistor.

The advantage of adding this to your BOM over using a resistive divider is twofold: (1) very low power dissipation, which is the major flaw of resistors as a divider; and (2) it maintains an even division of the rail, regardless of load current for each polarity. The latter is the real issue with onboard electronics — especially as batteries get weak.

However, even if you still wanted to go with two batteries for two polarities, then I would put them outside the guitar and have a jack mounted through the rear cover plate to connect the power to the PCB.

There are some neat four-cell AA battery boxes that are about the width of a wide leather guitar strap. Just re-wire the box as two sets of two batteries and you have a ±3V power supply.

(FYI, contrary to your assertion, there is no headroom issues with a ±3V supply, considering the low output level of the pickups).

I have one such battery box that also has a belt clip — if you want to wear it instead of having it on the strap. Not sure where I bought mine, but there's a very nice one with a power jack, switch, and aluminum belt clip at **www.batteryspace.com/BatteryHolder10xAA5x2With526AWGWireLeads-RoHSCompliant-2.aspx**.

If you went with this box, I'd simplify things and go with a single polarity battery box (which preserves the nice power jack on this unit), and use the TI rail splitter IC on the PCB.

The beauty of the battery box scheme is that you can use a small diameter jacketed cable from the battery box to the back of the guitar, and it will not be intrusive.

Servicing the batteries during a gig becomes a non-issue. (Heck, you can even have a spare battery box in the wings, if needed.)

The only negative issue with this scheme is that you have to use a right angle style jack on the guitar's back plate or your body rubbing against the guitar body will eventually cause a cable failure.

Something like a stereo 3.5 mm right angle plug and cable can be stolen from a $3 pair of headphones for the task, which is nice since the plug is molded to the cable for strain relief. This is still a weakness in the overall power supply cabling method, however.

So, my best solution was to lose the local (to the guitar) power supply altogether. What I did instead was to change the guitar's 1/4" jack from a mono to a stereo version. Then, the cable from my guitar to my pedal board needs to become a stereo cable, as well.

However, this is not a big sacrifice and it's easy to keep a few spare cables in your gig bag in case one fails.

The purpose of the added conductor is to use the new ring terminal on the three-cond plug as the positive supply voltage to the guitar's internal PCB. The tip stays as the signal terminal.

With the power supply now off-loaded from the instrument, I never have a battery issue and I can get a well filtered and well regulated constant voltage to the PCB (since I have a very stiff home-brewed 9V power supply built into my pedal board).

This last scheme is a big architectural improvement, as it makes the roll-out of my guitar electronics agnostic to any guitar I own since the space issues of how to fit batteries into the instrument and keep them stably located are now gone.

Plus, with a line-powered DC source of supply, I will always have full voltage available to the pre-amp, making low rail dropout concerns a non-issue. (BTW, I keep a gel cell on trickle charge in parallel to the line powered 9V DC power supply, just in case the venue's power quality is poor.)

Another advantage: I also never have to open the instrument's control cavity again, which preserves the integrity of the wood body by not exercising the cavity cover's

wood mounting screws.

The only significant negative change — if you can call it that — is the need for a three-cond jumper from the guitar to the pedal board, but even this is not a big deal since there are ready-made stereo guitar cables available. Just Google it.

Aside: Since you have done the work to decipher the active filter calculations for your schematic, I wonder if you would be able to create a nice Excel-based macro that would allow one to change RC values in your schematic and see the effects on a plotted frequency response graph.

Admittedly, I have been lazy over the years and simply played with values until I heard what I liked. Since you have done most of the legwork, it would facilitate knowing how to tweak the design without explicitly doing SPICE models.

Finally, on the subject of shielding, I prefer to use a spray-in coating rather than copper foil. You will find the adhesive of the copper strips will not be permanent and so the shield itself will fall apart. Too many interfaces. (Doesn't stick too well to wood either!)

There's a good example of such a spray shielding material at **www.mgchemicals.com/products/protective-coatings/emi-rfi-shielding/total-ground-carbon-conductive-coating-838**.

Of course, you need to pretty much put the entire guitar inside a large plastic bag, carefully create a void in the area of the control cavity, and use painter's tape to attach the bag opening to the edge of the cavity.

For the plastic cavity cover, there are primers available for plastic. Use that first, and then spray the inside of the cover with the shielding spray. You will get adequate mating of the conductive surfaces of the cover to the cavity if you mask the cavity well.

Thanks for the great article.

Jon

Thanks for sharing your experiences of some alternatives to the guitar hotrod project. Having hung out with musicians my entire life, I just knew there would be different wants and needs, so tried to keep it general for modifications such as those you propose. The arrangement I ended up with suits my purposes perfectly, but there's no reason to expect it to match everyone else's needs as well. Long live customization!

We're all different, but I prefer to have the batteries within the guitar, obviating the need for a special cord, strap, or external pack.

Battery life has never been a problem, even back some 30 years ago when I was gigging and performing daily. I would usually replace the batteries every December 31st, whether they needed it or not. One of the guitars that I outfitted for a neighbor is on the same set of batteries these three years later. Opening the rear cavity so infrequently really isn't much of an inconvenience.

The conductive paint sounds interesting, but foil tape has worked very well for me. Just clean out the sawdust left by the manufacturer, swab the cavity with a cotton ball soaked in rubbing alcohol, and away you go. In any event, the entire cavity is filled with foam rubber making it impossible for the foil to come loose.

I suppose a person could devise an Excel worksheet for the frequency calculations (maybe my article, "Spreadsheets: The Forgotten Analog Design Tool," which appeared in the October 2006 issue of Nuts & Volts, pp. 59-63, will suggest something).

I sure found the SPICE approach to be fast and easy, however, and the visual representation of the frequency response via Bode plots is a definite plus.

Your suggestions offer a number of excellent options many readers will find useful as they personalize the project to their own needs. Thanks again for contributing them!

Thomas Henry

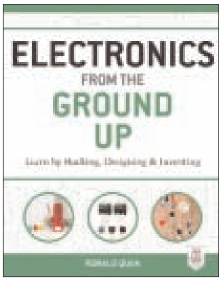
For complete product details, visit our webstore!!

The Nuts & Volts WEBSTORE

GREAT FOR DIYers!

Electronics from the Ground Up: Learn by Hacking, Designing, and Inventing

by Ronald Quan



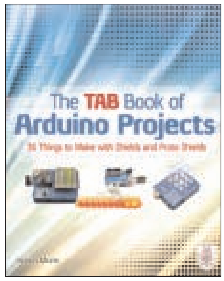
Are you fascinated by the power of even the smallest electronic device? Electronics from the Ground Up guides you through step-by-step experiments that reveal how electronic circuits function so you can advance your skills and design custom circuits. You'll work with a range of circuits and signals related to optical emitters and receivers, audio, oscillators, and video.

Paper back 544 pages.

Price $30.00

The TAB Book of Arduino Projects

by Simon Monk



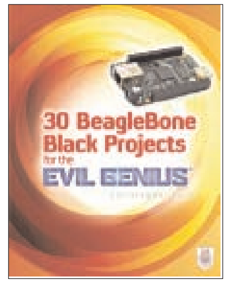
The ultimate collection of DIY Arduino projects!

In this easy-to-follow book, electronics guru Simon Monk shows you how to create a wide variety of fun and functional gadgets with the Arduino Uno and Leonardo boards. Filled with step-by-step instructions and detailed illustrations, The TAB Book of Arduino Projects: 36 Things to Make with Shields and Proto Shields provides a cost estimate, difficulty level, and list of required components for each project.

Regular Price $30.00 Sale Price $23.95

30 BeagleBone Black Projects for the Evil Genius

by Christopher Rush



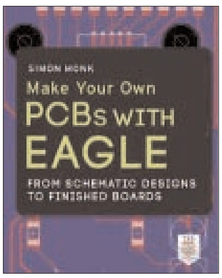
Fiendishly Fun Ways to Use the BeagleBone Black!

This wickedly inventive guide shows you how to program and build fun and fascinating projects with the BeagleBone Black. You'll learn how to connect the BeagleBone Black to your computer and program it, quickly mastering BoneScript and other programming tools so you can get started right away.

$25.00

Make Your Own PCBs with EAGLE

by Eric Kleinert



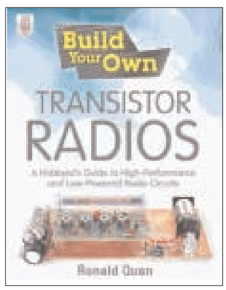
Featuring detailed illustrations and step-by-step instructions, *Make Your Own PCBs with EAGLE* leads you through the process of designing a schematic and transforming it into a PCB layout. You'll then move on to fabrication via the generation of standard Gerber files for submission to a PCB manufacturing service. This practical guide offers an accessible, logical way to learn EAGLE and start producing PCBs as quickly as possible.

$30.00

Build Your Own Transistor Radios

by Ronald Quan

A Hobbyist's Guide to High Performance and Low-Powered Radio Circuits



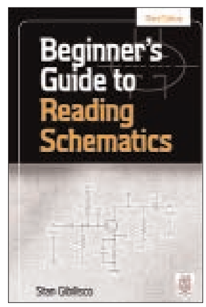
Create sophisticated transistor radios that are inexpensive yet highly efficient. Inside this book, it offers complete projects with detailed schematics and insights on how the radios were designed. Learn how to choose components, construct the different types of radios, and troubleshoot your work.

Paperback, 496 pages

$49.95

Beginner's Guide to Reading Schematics, 3E

by Stan Gibilisco

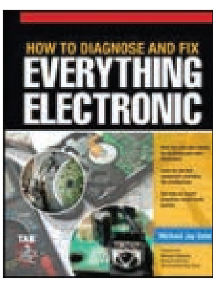


Navigate the roadmaps of simple electronic circuits and complex systems with help from an experienced engineer. With all-new art and demo circuits you can build, this hands-on, illustrated guide explains how to understand and create high-precision electronics diagrams. Find out how to identify parts and connections, decipher element ratings, and apply diagram-based information in your own projects.

$25.00

How to Diagnose and Fix Everything Electronic

by Michael Jay Geier



Master the Art of Electronics Repair

In this hands-on guide, a lifelong electronics repair guru shares his tested techniques and invaluable insights. *How to Diagnose and Fix Everything Electronic* shows you how to repair and extend the life of all kinds of solid-state devices, from modern digital gadgetry to cherished analog products of yesteryear.

$24.95

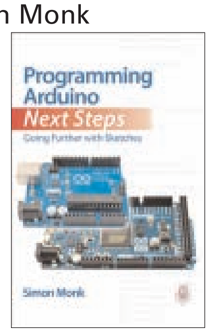
Programming PICs in Basic

by Chuck Hellebuyck

If you wanted to learn how to program microcontrollers, then you've found the right book! Microchip PIC microcontrollers are being designed into electronics throughout the world and none is more popular than the eight-pin version. Now the home hobbyist can create projects with these little microcontrollers using a low cost development tool called the CHIPAXE system and the Basic software language. Chuck Hellebuyck introduces how to use this development setup to build useful projects with an eight-pin PIC12F683 microcontroller. **$14.95**

Programming Arduino Next Steps: Going Further with Sketches

by Simon Monk

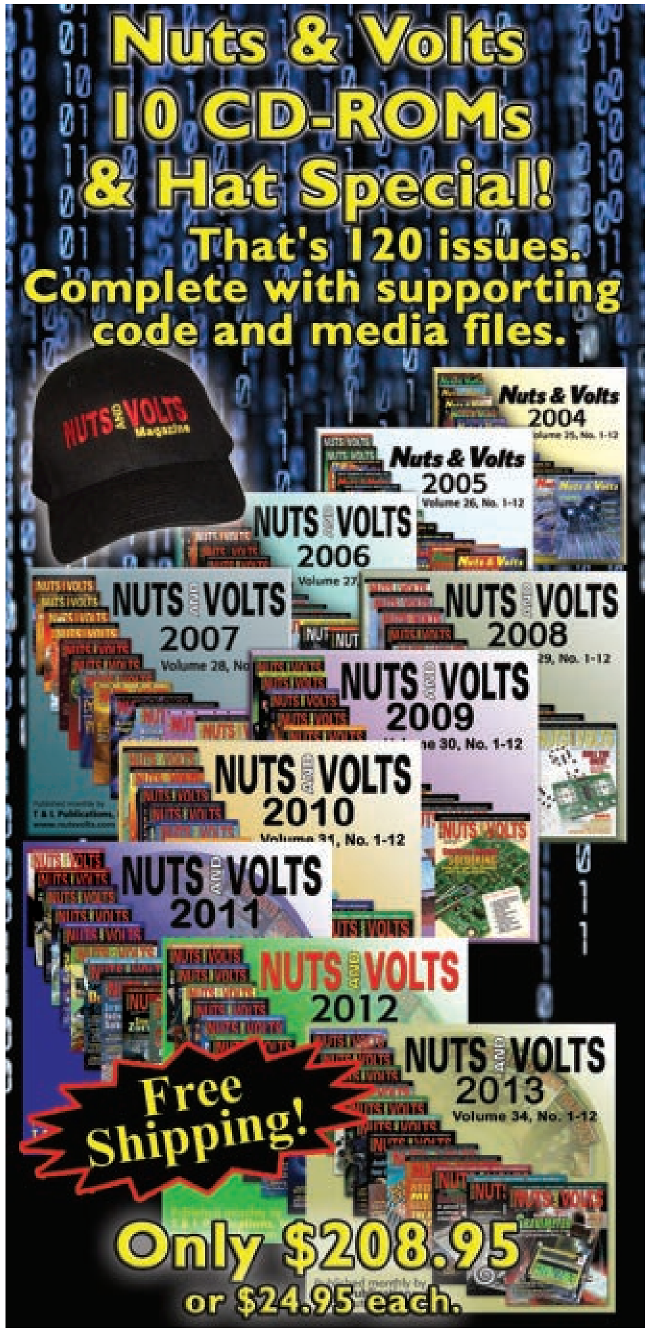


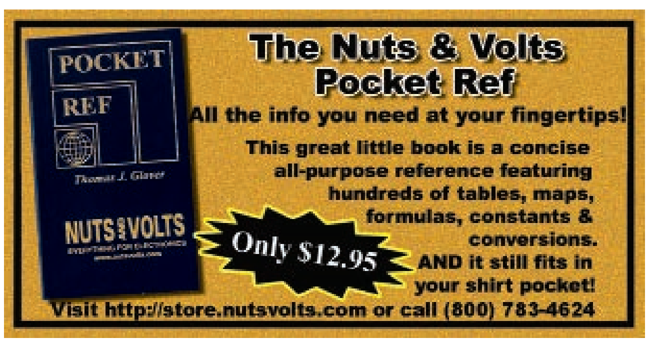
In this practical guide, electronics guru Simon Monk takes you under the hood of Arduino and reveals professional programming secrets. Also shows you how to use interrupts, manage memory, program for the Internet, maximize serial communications, perform digital signal processing, and much more. All of the 75+ example sketches featured in the book are available for download. **$20.00**

Order online @ www.store.nutsvolts.com
Or CALL 1-800-783-4624 today!

EDUCATIONAL

CD-ROM SPECIAL



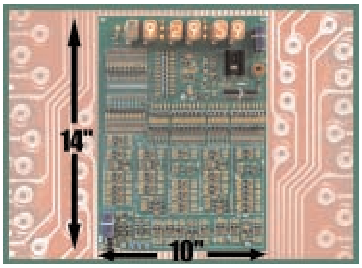


GIFTS FOR GEEKS

CALL 1-800-783-4624 today!
Order online @ www.nutsvolts.com

PROJECTS

Neon Transistor Clock Kit

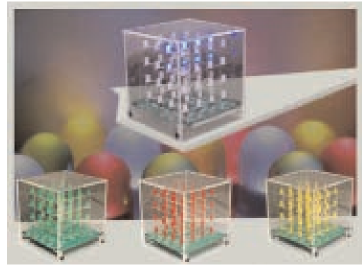


Add HIGH VOLTAGE to your clock!
This is a Nixie Tube display version of the Transistor Clock. It uses only discrete components — no integrated circuits. Approximatly 1,250 of them!
This kit is a "must have" for the analog electronic hobbyist.
For more info, see the April 2012 issue or download the 87 page manual from our webstore.
Sale Price **$245.95**

Peek-a-boo Ghost Kit

The peek-a-boo ghost kit is a fun, low cost multi-use microcontroller kit. When triggered by the included motion sensor, this mini animatronic waves its arms, lights its LED eyes, and plays back the sounds you record. Perfect for kids, this kit can be used to create a fun Halloween prop for your desk, front door, or walkway. Watch the video to see this cool kit in action. Available in both a program-it-yourself or with a pre-programmed PICAXE chip option.
Un-programmed Chip Kit **$29.95**
Pre-programmed Chip Kit **$37.95**

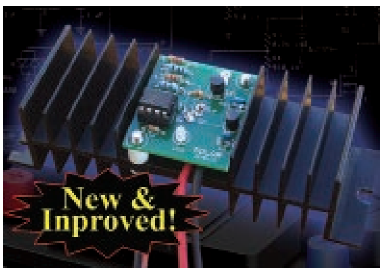
3D LED Cube Kit



This kit shows you how to build a really cool 3D cube with a 4 x 4 x 4 monochromatic LED matrix which has a total of 64 LEDs. The preprogrammed microcontroller that includes 29 patterns that will automatically play with a runtime of approximately 6-1/2 minutes.
Colors available: Green, Red, Yellow & Blue

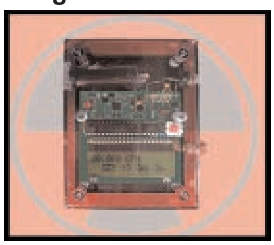
$57.95

Solar Charge Controller Kit 2.0



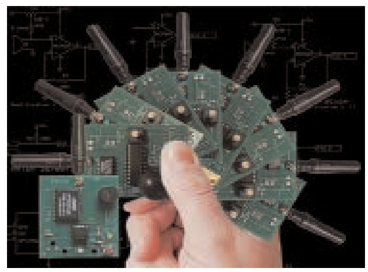
If you charge batteries using solar panels, then you can't afford not to have them protected from over-charging. This 12 volt/12 amp charge controller is great protection for the money. It is simple to build, ideal for the novice, and no special tools are needed other than a soldering iron and a 9/64" drill!
$27.95

Geiger Counter Kit



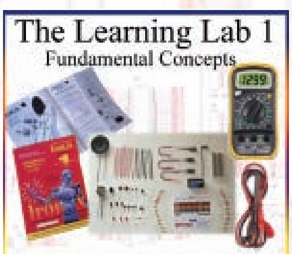
This kit is a great project for high school and university students The unit detects and displays levels of radiation, and can detect and display dosage levels as low as one micro-roentgen/hr. The LND 712 tube in our kit is capable of measuring alpha, beta, and gamma particles.
Partial kits also available.
$159.95

Super Detector Circuit Set

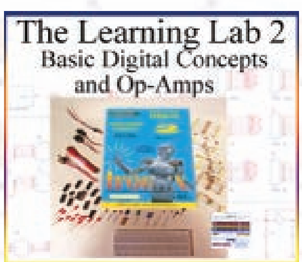


Pick a circuit!
With one PCB you have the option of detecting wirelessly: temperature, vibration, light, sound, motion, normally open switch, normally closed switch, any varying resistor input, voltage input, mA input, and tilt, just to name a few.
$32.95

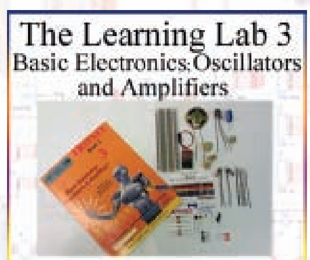
FOR BEGINNER GEEKS!



$59.95



$49.95



$39.95

The labs in this series — from GSS Tech Ed — show simple and interesting experiments and lessons, all done on a solderless circuit board.

As you do each experiment, you learn how basic components work in a circuit, and continue to build your arsenal of knowledge with each successive experiment.

For more info and a promotional video, please visit our webstore.

CLASSIFIEDS

THE OWNERSHIP, MANAGEMENT, AND CIRCULATION STATEMENT OF NUTS & VOLTS MAGAZINE, Publication Number: 1528-9885 is published monthly. Subscription price is $26.95. 7. The complete mailing address of known office of Publication is T&L Publications, Inc., 430 Princeland Ct., Corona, Riverside County, CA 92879-1300. Contact Person: Larry Lemieux. Telephone: (951) 371-8497. 8. Complete Mailing address of Headquarters or General Business Office of Publisher is T&L Publications, Inc., 430 Princeland Ct, Corona, CA 92879. 9. The names and addresses of the Publisher, and Associate Publisher are: Publisher, Larry Lemieux, 430 Princeland Ct., Corona, CA. 92879; Associate Publisher, Robin Lemieux, 430 Princeland Ct., Corona, CA 92879. 10. The names and addresses of stockholders holding one percent or more of the total amount of stock are: John Lemieux, 430 Princeland Ct., Corona, CA 92879; Lawrence Lemieux, 430 Princeland Ct., Corona, CA 92879; Audrey Lemieux, 430 Princeland Ct., Corona, CA 92879. 11. Known Bondholders, Mortgagees, and other security holders: None. 12. Tax Status: Has not changed during preceding 12 months. 13. Publication Title: Nuts and Volts 14. Issue Date for Circulation Data: October 2013-September 2014. 15. The average number of copies of each issue during the proceeding twelve months is: A) Total number of copies printed (net press run); 33,897 B) Paid/Requested Circulation (1) Mailed Outside County subscriptions: 22,011 (2) Mailed In-County subscriptions: 0 (3) Paid Distribution Outside the Mail including Sales through dealers and carriers, street vendor, and counter sales and other paid distribution outside USPS: 5,906 (4) Paid Distribution by other classes of mail through the USPS: 0; C) Total Paid Distribution: 27,917; D) Free or Nominal Rate Distribution by mail and outside the mail (1) Free or Nominal Rate Outside-County Copies: 0 (2) Free or Nominal Rate In-County Copies: 0 (3) Free or Nominal Rate Copies Mailed at other classes through the USPS: 0 (4) Free or Nominal Rate Distribution Outside the mail: 1,660; E) Total Free or Nominal Rate Distribution: 1,660; F) Total Distribution: 29,576; G) Copies not distributed: H) Total: 4,320; Percent paid circulation: 94.39%. Actual number of copies of the single issue published nearest the filing date is September 2013; A) Total number of copies printed (net press run) 33,690; B) Paid/Requested Circulation (1) Mailed Outside County subscriptions: 21,509 (2) Mailed In-County subscriptions: 0 (3) Paid Distribution Outside the Mail including Sales through dealers and carriers, street vendor, and counter sales and other paid distribution outside USPS: 6,087 (4) Paid Distribution by other classes of mail through the USPS: 0; C) Total Paid Distribution: 27,596; D) Free or Nominal Rate Distribution by mail and outside the mail (1) Free or Nominal Rate Outside-County Copies: 0 (2) Free or Nominal Rate In-County Copies: 0 (3) Free or Nominal Rate Copies Mailed at other classes through the USPS: 0 (4) Free or Nominal Rate Distribution Outside the mail: 2,415; E) Total Free or Nominal Rate Distribution: 2,415; F) Total Distribution: 30,011; G) Copies not distributed: 3,679; H) Total: 33,690; Percent paid circulation: 91.95%. I certify that these statements are correct and complete. Lawrence Lemieux, Publisher - 10/06/2014.

TECH FORUM
READER-TO-READER

>>> QUESTIONS

D Cell or Gell Cell Adapter for Nikon Cameras

I have the following three Nikon cameras: D3200, D5100, and D5300 that I want to use in my black bear wildlife research work. There are no electrical outlets to recharge the small lithium-ion batteries that come with the cameras in the desolate areas that I frequent. A car inverter to a regular lithium charger is out of the question because four-wheel drives can't get back to the remote areas that are accessible only by horse and by packing on foot.

Furthermore, the cameras need to be in the "on" mode for several days at a time and left unattended in cave areas, where continually re-entering the caves would not be particularly safe were I to repeatedly reinstall the little batteries. There is some flash power being consumed as well with every exposure, so the batteries drain fairly rapidly.

I would like to run a bipolar cable from a couple of parallel connected 12 volt/18 amp-hour batteries into a drilled hole in the camera battery snap cover in the camera base. That probably will probably require cutting open and possibly destroying a battery pack and removing the lithium-ion cells so as to connect the voltage dropped wires to the battery terminals inside the battery pack which, in turn, will make contact with the camera's internal fixed contacts.

Can you please assist me with a "camera safe" regulated circuit to drop the 12 volt gell cells to a constant regulated output level for a Nikon EN-EL14 lithium-ion battery which is rated at 7.4 volts at 1,050 mAh (7.8 Wh)? The Nikon MH-24 battery charger that comes with the cameras has a charger output of 8.4 volts at 0.9 amps and charges each EN-EL14 in about 2.5 hours. I would greatly appreciate your help and advice with a voltage dropping circuit and battery disassembly details, or if disassembly is not required, how to proceed with the hook-up.

#11141 **John Graff**
via email

Brushless Fans and Solar Panels

I have three 12 VDC brushless fans and am considering running them from a 12V 30W solar panel. Two of the fans are rated at 5.4W and one at 7.6W. Unfortunately, the brushless motors can tolerate a maximum voltage of 13.8V and the solar panel has an open circuit voltage of over 17V. I am afraid I could fry the electronics in the fans with this solar panel but I can't find a 12V solar panel that outputs no more than 12V. I am sure this is not the first time this problem has occurred. What options do I have?

#11142 **Terry Tanber**
San Diego, CA

QuickBASIC Program

Since you have various articles dedicated to programming PICs in Basic as well as coding for Arduino, I thought that maybe a proficient *N&V* reader could help me with a sorting program by correctly coding in Microsoft QuickBASIC.

What I'd like to accomplish is by electing choice (**1**), sort in ascending order just the six-numbered string, paired with the correct and correspoding "date" and "odd/even" strings. Therefore, the six-numbered string is sorted in ascending order while the corresponding date and odd/even string must stay with that six-numbered string (the dates and odd/even strings must also be swapped).

Also, I'd like to print to the screen — bit by bit/segment by segment — the entire generated output by depressing any keyboard button to advance to the next screen segment until I reach the end of the generated sortition. The way I've coded it, it just goes automatically to the last part of the screen output, without being able to ever visualize either the start nor the middle of the generated screen output.

As for choice (**2**), I'd like to only sort the string dates — in ascending order — paired with the correct six-numbered strings and corresponding odd/even strings. Lastly, I'd like to print to screen (just as for choice **1** — bit by bit) by depressing some button in order to advance to the next screen segment, until the entire generated sortition would have appeared on the screen.

I'm planning to add more data lines in the future, where I'll have to update lines 200 and 390.

Listing 1 indicates what I'd like. I learned QuickBASIC in the early '80s. My coding partially works, but I can't remember how to correctly do everything at my seasoned age of 56.

#11143 **Don Franklin**
via email

All questions *AND* answers are submitted by *Nuts & Volts* readers and are intended to promote the exchange of ideas and provide assistance for solving technical problems. All submissions are subject to editing and will be published on a space available basis if deemed suitable by the publisher. Answers are submitted by readers and **NO GUARANTEES WHATSOEVER** are made by the publisher. The implementation of any answer printed in this column may require varying degrees of technical experience and should only be attempted by qualified individuals.

Always use common sense and good judgment!

>>> ANSWERS

[#9143 - September 2014]

Noise Cancelling

Is there such a thing as a noise cancelling technology that would allow me to play my guitar without disturbing the neighbors?

I get complaints about noise from my apartment when I play. I just can't

Send all questions and answers by email to **forum@nutsvolts.com** or via the online form at **www.nutsvolts.com/tech-forum**

use headphones and I've tried putting blankets on the wall for sound-proofing, but I still get complaints. How about an electronic solution?

There is no practical noise cancelling technology on that scale. Building an anechoic chamber would be very expensive. Playing guitar in a vacuum would not allow you to hear yourself (since sound does not travel in a vacuum) and would suffocate you.

```
1 CLS
192 REM:
*******************************************************************************************************
193 REM: (1) SORT IN ASCENDING ORDER: ALL ACTUAL NUMBERS DRAWN (and the corresponding 'dates drawn')
194 REM: (2) SORT IN ASCENDING ORDER: DATES DRAWN (and the corresponding 'numbers drawn')
195 REM:
*******************************************************************************************************
196 REM: C$ = RATIO OF EVEN/ODD NUMBERS CONTAINED
197 REM: B$ = ACTUAL NUMBER DRAWN
198 REM: N$ = DATE DRAWN
199 REM: *****************************************************************
200 DIM N$(200), B$(200), C$(200)
201 REM: ******************************
210 REM - READ number of records
211 REM: ******************************
220 READ K
221 REM: *********************************
230 REM - READ records into ARRAYS
231 REM: *********************************
240 FOR I = 1 TO K
250 READ N$(I), B$(I), C$(I)
255 NEXT I
260 PRINT "****************************************************************************"
261 PRINT "SORT based on ALL ACTUAL NUMBERS DRAWN by ENTERing (1) OR "
262 PRINT "SORT based on DATES DRAWN by ENTERing (2) ";
263 INPUT C
264 IF C <> 1 AND C <> 2 THEN PRINT "---- < OOPS!.WRONG CHOICE ENTERED!.TRY AGAIN!. ENTER either (1) or (2)!. >
----": GOTO 260
265 IF C = 1 THEN 270
266 IF C = 2 THEN 345
267 REM: *********************************************************************************
268 REM : SORT based on ALL ACTUAL NUMBERS DRAWN by ENTERing (1)
269 REM: *********************************************************************************
270 FOR I = 1 TO K
280 FOR J = 1 TO K - 1
290 IF B$(J) > B$(J + 1) THEN SWAP B$(J), B$(J + 1)
300 NEXT J
310 NEXT I
311 D$ = N$(I) + " , " +B$(I) + ", " + C$(I)
320 REM - sorted in ascending order: ACTUAL NUMBERS DRAWN (with corresponding 'dates drawn') to be printed
330 FOR I = 1 TO K
335 PRINT I, D$
339 NEXT I
340 GOTO 390
342 REM: **********************************************************
343 REM: SORT based on DATES DRAWN by entering (2)
344 REM: **********************************************************
345 FOR I = 1 TO K
346 FOR J = 1 TO K - 1
347 IF N$(J) > N$(J + 1) THEN SWAP N$(J), N$(J + 1)
348 NEXT J
349 NEXT I
350 REM - sorted in ascending order: DATES DRAWN (with corresponding 'numbers drawn') to be printed
355 FOR I = 1 TO K
360 PRINT I, N$(I), B$(I), C$(I)
365 NEXT I
390 DATA 200
10001 DATA "10/22/1994", "10,16,20,28,39,40", "(5Even/1Odd)"
10002 DATA "12/10/1994", "21,23,35,37,39,48", "(1Even/5Odd)"
10003 DATA "01/07/1995", "24,31,33,35,40,44", "(3Even/3Odd)"
10004 DATA "02/04/1995", "11,20,21,24,29,42", "(3Even/3Odd)"
10005 DATA "02/11/1995", "14,17,25,30,31,34", "(3Even/3Odd)"
10006 DATA "05/06/1995", "15,18,23,24,36,48", "(4Even/2Odd)"
10007 DATA "05/20/1995", "11,27,34,35,37,45", "(2Even/4Odd)"
.......................................................................................................
10196 DATA "06/25/2003", "10,16,31,35,38,43", "(3Even/3Odd)"
10197 DATA "06/28/2003", "18,19,26,34,46,47", "(4Even/2Odd)"
10198 DATA "07/02/2003", "20,25,26,36,37,47", "(3Even/3Odd)"
10199 DATA "07/16/2003", "10,16,22,24,26,37", "(5Even/1Odd)"
10200 DATA "08/23/2003", "17,24,26,28,41,44", "(4Even/2Odd)"
40000000 END
```

Listing 1

That leaves two possible options.

Relax your ban on headphones and use a product such as the $40 Amplug — the headphone guitar amp that lets you enjoy serious guitar sound, fast. See **www.voxamps.com/amplug** or possibly go with the low tech solution of playing under a wool blanket.

More than 40 years ago, we wanted to record short skits for radio brodcast in our college dorm room. Our input was not as loud as an accoustic guitar, and we were close mic-ed for voice and sound effects. We had such good sound isolation, that it effectively blocked the external noise of nearby stereos, running feet, shouting, and anti Vietnam War riots outside the building. Many people wondered where we found such a quiet place to record! This probably will not work for amplified electric guitars. You could use a self-standing tent, covering it with heavy blankets down to the floor. The level outside the tent will be dramatically attenuated.

Barry Cole
Lacey, WA

[#10143 - October 2014]
Carbon or Metal?

Is there a rule of thumb for when it is better to use carbon film resistors over metal film resistors?

As a "rule of thumb," I recommend: Choose carbon film resistors when low cost (at high quantity) is the main consideration. Use metal film resistors when low noise/low temperature coefficient, and/or greater precision are important.

All types of resistors generate thermal noise — also called "Johnson noise" — which increases as resistance and temperature increase. This comes directly from the laws of physics which, as Engineer Scotty famously said on Star Trek, "you cannae change." However, current flowing through a resistor causes additional noise — including "shot noise" — and

this is greater in carbon film than in metal film types. That is why the latter are preferred for critical analog circuitry such as audio preamplifiers. Resistor noise hardly matters in digital applications.

For more information about noise, see Joe Geller's excellent resistor noise measurement project (JCan) in the July 2007 *Nuts & Volts*. After building the JCan kit, I could easily measure more noise in carbon composition types compared to metal film resistors of equal value when current was applied. There was no difference with zero current — just the thermal noise "background." The noise difference between carbon film and metal film resistors under power was smaller, but detectable.

The temperature coefficient tells you how much a resistor's value can change with temperature in units of parts per million (PPM) per °C or °K. Common axial lead (through hole) 1/4 watt metal film resistors are rated 50 or 100 PPM/°C, but their carbon film cousins are 350 to 700 PPM/°C. Sensitivity to temperature is a consideration in many sensor and measurement applications, but it does not matter in digital circuits.

Assuming the resistor offerings at Mouser Electronics (**www.mouser.com**) are typical of the industry, the vast majority of 1/4 watt axial units are available only at one percent tolerance for metal film, and five percent for carbon film. Even if I wanted one, I could not buy a 5% metal film or a 1% carbon film resistor. Therefore, choice of type is related to precision for practical purposes. The cost difference is negligible for small quantities; you are mainly paying for the labor to count them out and bag them.

Clark Huckaby
via email

[#10144 - October 2014]
Chinese Capacitor Problem

I have a Viewsonic model VG2230WM LCD monitor that has quit. I read that these monitors have the "Chinese capacitor problem" but I can't figure out what exactly that means and what to do to fix it. Do I just replace all the caps on the PCB?

#1 Often, a monitor goes dark because the backlight power supply fails. Look for the power module which has the pink and blue wire backlight voltage connectors. Usually, there are two or four connectors for the backlight power. The capacitors on this module often fail due to heat over a period of several years, and they sometimes have bulging tops making them easy to spot. Replace all of the electrolytic capacitors on the power module; there may be five or six of them. In many cases, this will restore operation of the monitor.

Bill Seabrook, WETA-TV
via email

#2 I have repaired several flat screen monitors: one of them was the VG2230WM. I was able to find the power supply online and replace it after getting it open. Another one was fixed by buying the capacitor kit online and replacing each of them. The hardest part of any flat screen repair is getting them open without damage. Good luck.

Bob Smith
Prescott, AZ

Photoresistor Switcher

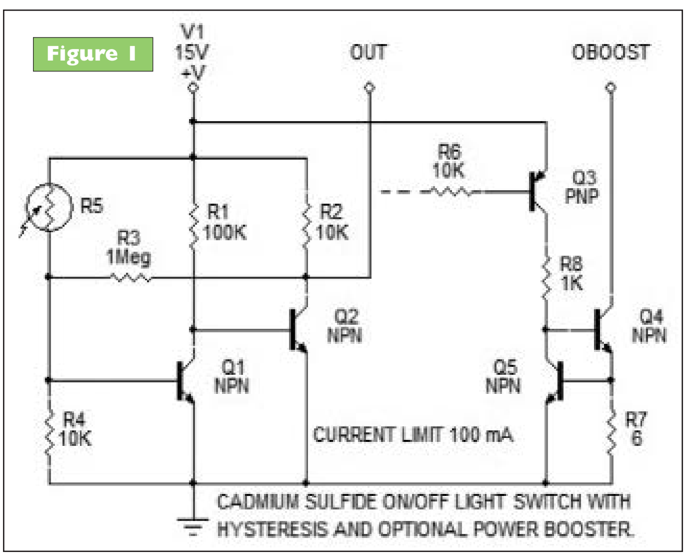
I would like to know if I have correctly connected the photoresistors (CdS photocells) shown in the provided diagram (printed last month) in order to turn OFF during the day and to turn ON during the night the two LED circuits attached to them. (If not, please indicate by a new diagram.)

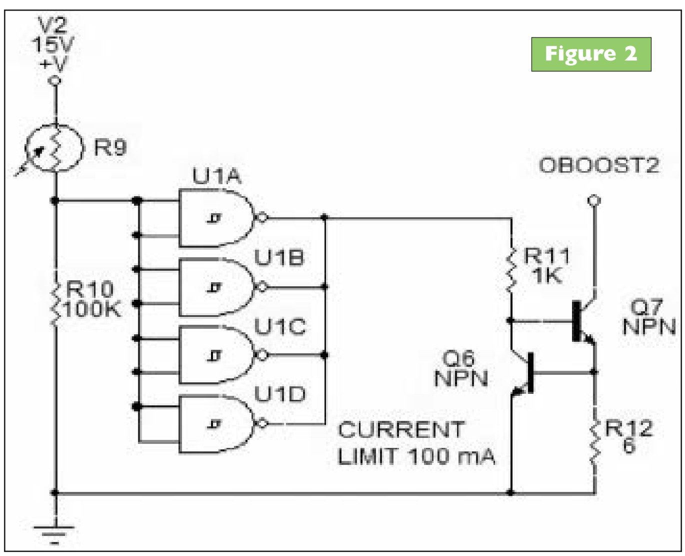
Also, I would like to know:

***a)** If any photoresistor would work.*

***b)** What would the optimum dark/light resistance values be for such a photoresistor?*

***c)** How would I calculate the values (any formula?) from the transistor side (2N2222) that would best fit this*





ON/OFF photoresistor switcher?

Depending on the current required by the different LED loads, the value of the Cds sensor is critical. In addition, you should include hysteresis to snap the load ON or OFF, rather than have it operate erratically at the dawn or dusk threshold. If the Cds sensor has a large value at the critical threshold, there will not be enough current to bias the transistor ON to its saturation state.

To remedy this problem, I have included two circuits which solve your issues. **Figure 1** uses two transistors with positive feedback to create the needed hysteresis and gain to drive an output stage that provides 100 mA current limiting in case your load circuit shorts out.

Figure 2 uses a 4093 quad-nand Schmitt trigger (that has one volt hysteresis) and drives the same output stage as the first circuit. The second circuit is probably the easiest one to implement, but requires the 14-pin 4093 IC. You will still need to select a Cds cell that has around a megohm of resistance when dark and 10K or less resistance when light. Hope this helps you.

Ron Hoffman
Hoffman Electronics, Inc.
Solon, OH

LOOK FOR ■ SEARCH FOR ■ FIND

Find your favorite advertisers here!

ADvertiser INDEX

Get Ready For The Holidays With a Ramsey Kit!

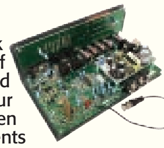
Super-Pro FM Stereo Radio Station Kit

- ✔ PLL synthesized for drift-free operation
- ✔ Built-in mixer - 2 line inputs and one microphone input, line level monitor output!
- ✔ Frequency range 88.0 to 108.0, 100 kHz steps
- ✔ Precision active low-pass "brick wall" audio filter!
- ✔ Dual LED bar graph audio level meters!
- ✔ Automatic adjustable microphone ducking!
- ✔ Easy to build through-hole design!

FALL FM SALE!

The true professional workhorse of our FM Stereo transmitter line, the FM100B has become the transmitter of choice for both amateurs and professionals around the world. From the serious hobbyist to churches, drive-in theaters, colleges and schools, it continues to be the leader. Not just a transmitter, the FM100B is a fully functional radio station and provides everything but the audio input and antenna system! Just add that and you're on the air!

This professional synthesized transmitter is adjustable directly from the front panel with a large LED digital readout of the operating frequency. Just enter the setup mode and set your frequency. Once selected and locked you are assured of a rock stable carrier with zero drift. The power output is continuously adjustable throughout the power range of the model selected. In addition, a new layer of anti-static protection for the final RF amplifier stage and audio inputs has been added to protect you from sudden static and power surges.



Audio quality is equally impressive. A precision active low-pass brick wall audio filter and peak level limiters give your signal maximum "punch" while preventing overmodulation. Two sets of rear panel stereo line level inputs are provided with front panel level control for both. Standard unbalanced "RCA" line inputs are used to make it simple to connect to the audio output of your computer, MP3 player, DVD player, cassette deck or any other consumer audio source. Get even more creative and use our K8094 below for digital storage and playback of short announcements and ID's! In addition to the line level inputs, there is a separate front panel microphone input.

All three inputs have independent level controls eliminating the need for a separate audio mixer! Just pot-up the source control when ready, and cross fade to the 2nd line input or mic! It's that simple! In addition to the dual stereo line inputs, a stereo monitor output is provided. This is perfect to drive studio monitors or local in-house PA systems. The FM100B series includes an attractive metal case, whip antenna and built in 110/220VAC power supply. A BNC connector is also provided for an external antenna. Check out our Tru-Match FM antenna kit, for the perfect mate to the FM100B transmitter. We also offer a high power kit as well as an export-only assembled version that provides a variable RF output power up to 1 watt. The 1 watt unit must utilize an external antenna properly matched to the operating frequency to maintain a proper VSWR to protect the transmitter.
(Note: The FM100B and FM100BEX are do-it-yourself learning kits that you assemble. The end user is responsible for complying with all FCC rules & regulations within the US or any regulations of their respective governing body. The FM100BWT is for export use and can only be shipped to locations outside the continental US, valid APO/FPO addresses or valid customs brokers for documented end delivery outside the continental US.)

FM100B	**Super-Pro FM Stereo Radio Station Kit, 5uW to 25mW Output**	*$239.95*
FM100BEX	**Super-Pro FM Stereo Radio Station Kit, 5uW to 1W Output**	*$299.95*

FALL FM SALE!

Digital Controlled FM Stereo Transmitters

- ✔ PLL synthesized for drift free operation
- ✔ Front panel digital control and display of all settings and parameters!
- ✔ Professional metal case for noise-free operation
- ✔ EMI filtering on audio and power inputs
- ✔ Super audio quality, rivals commercial broadcasts
- ✔ Available in domestic kit or factory assembled export versions

FALL FM SALE!

For more than a decade we've been the leader in hobbyist FM radio transmitters. We told our engineers we wanted a new technology transmitter that would provide FM100 series quality without the advanced mixer features. They took it as a challenge and designed not one, but TWO transmitters!

The FM30B is designed using through-hole technology and components and is available only as a do-it-yourself kit with a 25mW output very similar to our FM25 series. Then the engineers redesigned their brand-new design using surface mount technology (SMT) for a very special factory assembled and tested FM35BWT version with 1W output for our export only market! All settings can be changed without taking the cover off! Enter the setup mode from the front panel and step through the menu to make all of your adjustments. A two line LCD display shows you all the settings! In addition to the LCD display, a front panel LED indicates PLL lock so you know you are transmitting.

Besides frequency selection, front panel control and display gives you 256 steps of audio volume (left and right combined) as well as RF output power. A separate balance setting compensates for left/right differences in audio level. In addition to settings, the LCD display shows you "Quality of Signal" to help you set your levels for optimum sound quality. And of course, all settings are stored in non-volatile memory for future use! Both the FM30B and FM35BWT operate on 13.8 to 16VDC and include a 15VDC plug-in power supply. *(Note: After assembly of this do-it-yourself hobby kit, the user is responsible for complying with all FCC rules & regulations within the US, or any regulations of their respective governing body. FM35BWT is for export use and can only be shipped to locations outside the continental US or valid APO/FPO addresses or valid customs brokers for end delivery outside the continental US.)*

FM30B	**Digital FM Stereo Transmitter Kit, 0-25mW, Black**	*$169.95*
FM35BWT	**Digital FM Stereo Transmitter, Assembled, 0-1W, Black (Export ONLY)**	*$259.95*

FALL FM SALE!

The Holiday Spotlight!

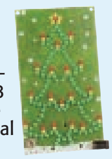
LED Animated Santa

This animated Santa and reindeer display has been our most popular holiday display for years! It contains a whopping 126 dazzling colored LEDs which make it a great holiday sign that is guaranteed to draw attention!

LED animated motion makes it come alive. Runs on standard 9V battery or 9-12VDC external power supply. Dazzle your friends this great display!

MK116 LED Animated Santa Kit $21.95

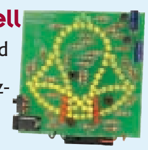
LED Christmas Tree



Electronic Christmas tree features 134 bright colored LEDs in the shape of a gorgeous holiday Christmas tree. Includes 18 random flashing blinking "candles" on the PC board! Runs on a 9V battery or external 9-12VDC power supply.

MK117 LED Christmas Tree Kit $24.95

LED Animated Holiday Bell



This PC board holiday bell is animated to simulate a bell swinging back and forth! 84 bright colored LEDs will dazzle you with holiday cheer! Includes an on/off switch. Runs on 9V.

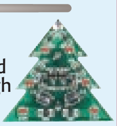
MK122 LED Animated Bell Kit $18.95

3D LED Christmas Tree

Not your average LED display! 4 branch sections give this tree a true 3D look! 16 red LEDs light it up with yellow LEDs for you to customize your tree! The base of the tree is actually the 9V battery acting as a self supporting base! Now that's pretty neat!

MK130 3D LED Christmas Tree Kit $7.95

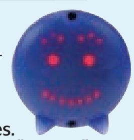
SMT LED Christmas Tree



Build this subminiature Christmas tree and learn SMT at the same time. Small enough to wear as a badge or pendant! Extra SMT parts are included so you can't go wrong! Runs on Li-Ion cell.

MK142 SMT LED Christmas Tree Kit $10.95

LED Animated Smiley



This cute little LED smiley displays 9 different animated expressions to suit any mood! You can choose between fixed or animated for a dynamic display. Features auto-power-off to save batteries. Runs on 3 standard AAA batteries. 3" x 3" x 1.28".

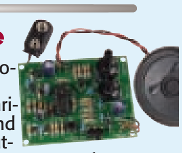
MK175 LED Animated Smiley Kit $19.95

LED Red Blinky

Alternately flashes two jumbo red LEDs. You've seen this used on name badges, hats, buttons, and attention getters for decades! Has been the most popular 1st learning kit for kids young and old! Runs on 3-9VDC.

BL1 LED Red Blinky Kit $7.95

Steam Engine & Whistle



Add the sound of a vintage locomotive steam engine and whistle to your holiday train set! Features variable engine speed and volume, and a whistle blast at the push of a button! Includes small speaker. Runs on 9VDC battery.

MK134 Steam Engine & Whistle Kit $11.95

ramsey
www.ramseykits.com

There's only so much room on these two pages, so check it all out in our new virtual electronic catalog! Flip through the pages and search with ease! Visit ***www.ramseycatalog.com***

Follow Us and SAVE $$

Follow us on your favorite network site and look for a lot of super deals posted frequently... exclusively for our followers!

Next Generation Classic Nixie Tube Clocks!

- ✔ Today's latest technology with yesterday's display!
- ✔ Choice of our signature hand rubbed Teak/Maple or acrylic base!
- ✔ Choice of Nixie tubes types (IN-14 or IN-82)
- ✔ Low-cost GPS receiver timebase option!
- ✔ Programmable color LED tube mood lighting!
- ✔ The ultimate conversation piece!

THE NEW STANDARD IN NIXIE CLOCKS!

Our next generation of classic Nixie tube clocks perfectly mesh today's technology with the Nixie era technology of the 60's. Of course, features you'd expect with a typical clock are all supported with the Nixie clock... and a whole lot more! Time wise, the clocks are designed around a quartz crystal timebase, therefore are not AC power frequency dependent like a lot of clocks. This means they can be used in any country regardless of power frequency, with the included 12VDC regulated power supply. The clocks are also programmable for 12 or 24 hour mode, various AM/PM indications, programmable leading zero blanking, and include a programmable alarm with snooze.

Unlike most Nixie clocks, the clocks also display the date in DD.MM.YY, MM.DD.YY, or YY.MM.DD format, which can be programmed to display for a few seconds at the end of each minute either as a static display, or by a neat scrolling in/out from alternating sides of the display. Display wise, the clocks feature a programmable night mode with dim or blank display, a programmable master blank tube saver, hard or soft fade digit change, and even have a built-in "Slot Machine" cathode poisoning prevention routine. Programming and setting the clock is a breeze with simple 2-button entries on the rear panel. The clocks are available in our signature hand rubbed Teak & Maple or futuristic clear acrylic bases.

WE CROSSED THE TECHNOLOGY TIMELINE!

We then jumped the technological timeline of the 60's Nixie displays by adding the latest multi-colored LEDs to the base of the Nixie tubes to provide hundreds of illumination colors to highlight the glass tubes! The LED lighting can be programmed to any color and brightness combination of the colors red, green, or blue to suit your mood or environment. Then we leaped over the technological timeline by integrating an optional GPS time base reference for the ultimate in clock accuracy! The small optional GPS receiver module is factory assembled and tested, and plugs directly into the back of the clock to give your Nixie clock accuracy you could only dream of! The new series clocks are available in 6-tube and 4-tube versions, with your choice of bases, and your choice of kit or factory assembled & tested. If you're looking for the ultimate conversation piece, with a trip down nostalgia lane, check out our clocks at www.ramseykits.com.

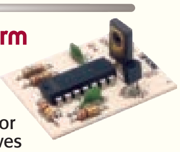
NIXIE CLOCKS **Classic Nixie Tube Clocks, Teak/Maple or See-Through Acrylic Base, Kit or Factory Assembled** **From $229.95**

RF Preamplifier

The famous RF preamp that's been written up in the radio & electronics magazines! This super broadband preamp covers 100 KHz to 1000 MHz! Unconditionally stable gain is greater than 16dB while noise is less than 4dB! 50-75 ohm input. Runs on 12-15 VDC.

SA7 **RF Preamp Kit** **$16.95**

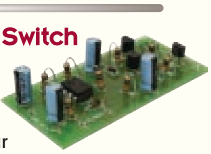
Mad Blaster Warble Alarm



If you need to simply get attention, the "Mad Blaster" is the answer, producing a LOUD ear shattering raucous racket! Super for car and home alarms as well. Drives any speaker. Runs on 9-12VDC.

MB1 **Mad Blaster Warble Alarm Kit** **$9.95**

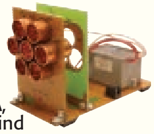
Automatic VOX Voice Switch



This popular VOX switch provides a switched output when it "hears" sound! Use it to key a transmitter, turn on lights, or anything... all controlled by your voice! Runs on 6-12VDC.

VS1 **Automatic VOX Switch Kit** **$9.95**

Air Blasting Ion Generator



Generates negative ions along with a hefty blast of fresh air, all without any noise! The steady state DC voltage generates 7.5kV DC negative at 400uA, and that's LOTS of ions! Includes 7 wind tubes for max air! Runs on 12-15VDC.

IG7 **Ion Generator Kit** **$64.95**

Passive Aircraft Monitor

PATENTED!

The hit of the decade! Our patented receiver hears the entire aircraft band without any tuning! Passive design has no LO, therefore can be used on board aircraft! Perfect for airshows, hears the active traffic as it happens! Available kit or factory assembled.

ABM1 **Passive Aircraft Receiver Kit** **$89.95**

Code Practice Oscillator

A great starter kit for young and old! Learn kit building while building a great little CW practice oscillator to learn the code! Built-in key, built in speaker, and audio tone adjust! Runs on a 9V battery.

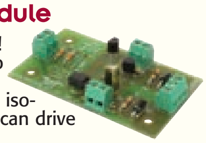
CPO3 **Code Practice Oscillator Kit** **$17.95**

Touch Switch

Touch on, touch off, or momentary touch hold, it's your choice with this little kit! Uses CMOS technology. Actually includes TWO totally separate touch circuits on the board! Drives any low voltage load up to 100mA. Runs on 6-12 VDC.

TS1 **Touch Switch Kit** **$9.95**

Optically Isolated Module



The hobbyist's headache solver! Converts any AC or DC signal to logic level. The beauty is that the input and output are totally isolated from each other! Output can drive up to 150mA at 40VDC.

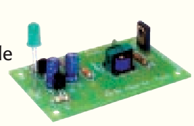
OM2 **Optically Isolated Module Kit** **$16.95**

Electronic Watch Dog

A barking dog on a PC board! And you don't have to feed it! Generates 2 different selectable barking dog sounds. Plus a built-in mic senses noise and can be set to bark when it hears it! Adjustable sensitivity... unlike my Greyhound and Boxer! 9-12VDC.

K2655 **Electronic Watch Dog Kit** **$39.95**

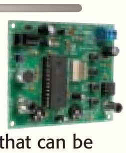
Tickle-Stick Shocker



The kit has a pulsing 80 volt tickle output and a mischievous blinking LED. And who can resist a blinking light and an unlabeled switch! Great fun for your desk, "Hey, I told you not to touch!" Runs on 3-6 VDC.

TS4 **Tickle Stick Kit** **$9.95**

Audio Recorder & Player



Record and playback up to 8 minutes of messages from this little board! Built-in condenser mic plus line input, line & speaker outputs. Adjustable sample rate for recording quality. 4-switch operation that can be remote controlled! Runs on 9-12VDC at 500mA.

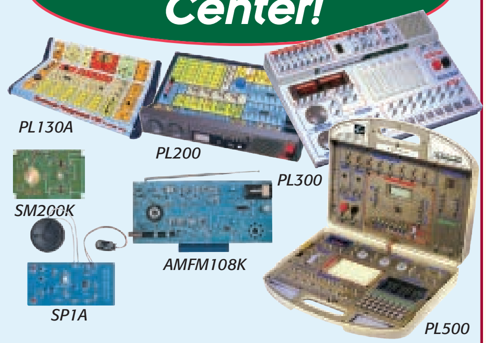
K8094 **Audio Recorder/Player Kit** **$32.95**

Precision PC Plane Antennas

Our LPY series PC antennas continue to be the favorite for virtually all RF and wireless applications. From microwave links, wireless mics, to RFID, we've got you covered. Check our site for details!

LPYSeries **Precision PC Plane Antennas** **from $29.95**

The Learning Center!



Fun Electronic Learning Labs

- ✔ Learn and build!
- ✔ 130, 200, 300, & 500 in one labs!
- ✔ Practical through hole and SMT soldering labs!
- ✔ Integrated circuit AM/FM radio lab!
- ✔ Super comprehensive training manuals!

Starting out our "All in One" series, the PL130A, gives you 130 different electronic projects, together with a comprehensive 162 page learning manual. A great start for the kids...young and old! Next, check out the PL200, that gives you 200 very creative and fun projects, and includes a neat interactive front panel with 2 controls, speaker, LED display and a meter. From there, step up to our PL300, which gives you 300 separate electronic projects along with s 165 page learning and theory manual. The PL300 walks you through the learning phase of digital electronics. If you're looking for the ultimate lab kit, check out our PL500. It includes a whopping 500 separate projects, a 152 page starter course manual, a 78 page advanced course manual, and a 140 page programming course manual! The PL500 covers everything from the basics to digital programming!

If you are looking to either learn or hone up on your through hole or SMT soldering skills check our SP1A and SM200K Practical Soldering Labs. You will be a soldering master in no time!

We make it easy to learn IC's while at the same time, building a neat AM/FM radio with our AMFM108K AM/FM IC lab kit. You will have a blast AND learn!

PL130A	**130-In-One Lab Kit**	**$39.95**
PL200	**200-In-One Lab Kit**	**$84.95**
PL300	**300-In-One Lab Kit**	**$109.95**
PL500	**500-In-One Lab Kit**	**$249.95**
SP1A	**Through Hold Soldering Lab**	**$9.95**
SM200K	**SMT Practical Soldering Lab**	**$22.95**
AMFM108K	**AM/FM IC Lab Kit & Course**	**$36.95**

GET THE NUTS & VOLTS DISCOUNT!

*Mention or enter the coupon code **NVRMZ142** and receive 10% off your order!*

800-446-2295

www.ramseykits.com

Prices, availability, and specifications are subject to change. According to Robin we are not responsible for typos, stupids, printer's bleed, or tryptophan sleep comas from the turkey, and she certainly writes the rules! Visit www.ramseykits.com for the latest pricing, specials, terms and conditions. Copyright 2014 Ramsey Electronics®... so there!

RAMSEY ELECTRONICS®
590 Fishers Station Drive
Victor, NY 14564
(800) 446-2295
(585) 924-4560